

AN ALGORITHM FOR OBTAINING SYNTHESIS EQUATIONS  
FOR ANY (MAXIMUM EIGHT LINKS) SINGLE DEGREE  
PLANAR MECHANISM FOR PATH GENERATION AND  
RIGID BODY GUIDANCE TYPES OF  
MOTION PROGRAMS

By

SYED ABDUL AZEEZ

Bachelor of Engineering

Osmania University

Hyderabad, India

1973

Submitted to the Faculty of the Graduate College  
of the Oklahoma State University  
in partial fulfillment of the requirements  
for the Degree of  
MASTER OF SCIENCE  
May, 1976

Thesis  
1974  
A993a  
cop. 2

AUG 26 1976

AN ALGORITHM FOR OBTAINING SYNTHESIS EQUATIONS  
FOR ANY (MAXIMUM EIGHT LINKS) SINGLE DEGREE  
PLANAR MECHANISM FOR PATH GENERATION AND  
RIGID BODY GUIDANCE TYPES OF  
MOTION PROGRAMS

Thesis Approved:

*Armanam H. San.*

Thesis Adviser

*Larry D. Zuhle*

*R. L. Lowery*

*D. D. Durbin*

Dean of the Graduate College

947473

## ACKNOWLEDGEMENTS

I would like to express my sincere thanks to my adviser, Dr. A. H. Soni, whose encouragement and perseverance were responsible for the work reported in this thesis.

Thanks are also extended to my friends and colleagues, Avinash Patwardhan, John Vadasz, Jack Lee, M. N. Siddhanty, Randy Mayo, Asif Ali and Tain Chunsinipong for having made my stay here at Oklahoma State University a lively one.

I deeply appreciate Mr. Syed Hamid and Mr. M. N. Siddhanty having given me their valuable time in discussing this problem with me. The discussions were very stimulative and very helpful.

My deep appreciation to my father, sister and brothers for their love and moral support.

Thanks are also extended to the School of Mechanical and Aerospace Engineering and to the National Science Foundation (Grant No. ENG:72-04207-A01) for having supported me financially all through my stay at Oklahoma State University as a graduate assistant.

Finally, Ms. Charlene Fries deserves a special thanks for her expert typing of the rough draft on such short notice, and Mrs. Grayce Wynd for the final draft.

## TABLE OF CONTENTS

Chapter	Page
I. INTRODUCTION . . . . .	1
II. DERIVATION AND EXPLANATION OF DISPLACEMENT AND ROTATION MATRICES . . . . .	4
III. DEFINITION OF NOTATIONS AND TERMS FOR THE SYMBOLIC COMPUTER PROGRAM . . . . .	11
IV. INPUT INFORMATION REQUIRED FOR THE COMPUTER PROGRAM . .	16
LISTING OF THE INPUT DATA FOR THE EIGHT LINK MECHANISM OF FIGURE 7 . . . . .	22
V. SELECTION OF COUPLER LINK . . . . .	24
VI. OBTAINING SYNTHESIS EQUATIONS . . . . .	34
COMPUTER OUTPUT FOR THE EIGHT LINK MECHANISM OF FIGURE 7 . . . . .	47
VII. SUMMARY AND CONCLUSIONS . . . . .	52
BIBLIOGRAPHY . . . . .	53
APPENDIX - LISTING OF THE COMPUTER PROGRAM . . . . .	55

## LIST OF FIGURES

Figure	Page
1. Displacement and Rotation of a Planar Link . . . . .	6
2. Rotation of a Planar Link . . . . .	9
3. Kinematic Links . . . . .	13
4. An Eight Link Kinematic Chain and an Eight Link Mechanism	13
5. A Dyad . . . . .	13
6. Removal of a Dyad . . . . .	15
7. Comparison of Mechanisms . . . . .	20
8. Adding a Dyad to a Four Link Mechanism . . . . .	25
9. Dyads Added to a Four Link Mechanism . . . . .	27
10. An Eight Link Mechanism . . . . .	29
11. A Four Link Mechanism . . . . .	35
12. Figure for Explaining Procedure No. 11 . . . . .	39
13. Figure for Explaining Procedure No. 12 . . . . .	40
14. Group-2 Mechanism . . . . .	42
15. Group-3 Mechanism . . . . .	43
16. Group-1 Mechanism . . . . .	45

## CHAPTER I

### INTRODUCTION

The modern kinematics had its beginning with the noted German kinematician, Franz Reuleaux. He was named professor at the Polytechnicum in Zurich in the year 1856. Here he developed the basic concepts fundamental to a study of kinematics. His classical book, Theoretische Kinematik, written in 1875, was translated into English by Kennedy, and titled Kinematics of Machines. He introduced the idea of constrained motion and recognized that it depended on the form of the surfaces of contact of the adjacent parts. Each of the working surfaces he called an element. Reuleaux defined a mechanism as a kinematic chain of connected links, one link being fixed. He identified synthesis as a concept; it is design, the creation of something new. Kinematically speaking, it is the conversion of a motion idea into hardware. Synthesis is the opposite of analysis. In analysis, the existing mechanism is examined and the motion it generates is studied.

The whole problem of synthesis can be considered to be made up of three parts. These three parts of synthesis are (1) the type synthesis, (2) the number synthesis, and (3) the dimensional synthesis. In type synthesis, the different kinematic pairs, the links, are decided upon to build the mechanism. In number synthesis, the total number of links and the different lower and higher kinematic pairs to be used are decided upon so that constrained motion of the mechanism can be obtained. Using

the Gruebler's mobility criterion, the degree of freedom of the mechanism can be determined.

After having decided upon the different types of pairs, links, and the different numbers of them to be used, it remains to determine the lengths of the different links which will enable the mechanism to generate the specified motion. The determination of the proper link lengths is called the dimensional synthesis of the mechanism.

In the nineteenth century, dimensional synthesis of mechanisms using analytical methods was not possible because of the too cumbersome calculations involved. The graphical techniques were used in the synthesis of mechanisms. The Germans and the Russians studied the graphical techniques and developed them to a high degree of sophistication. The interest in the area of kinematics was not aroused in the English speaking countries until Professor De Jonge in the year 1942 made critical remarks on this subject.

It is seen that after the introduction of high speed digital computers, the interest in the area of kinematics took a new turn in this country. New sophisticated analytical techniques were developed; the solution to most of these methods was possible only by the use of computers. Soni and Harrisberger (1)\* recently revealed the fact that at present there are more than 12,000 publications in the kinematics alone.

There are numerous analytical methods that exist today to synthesize multi-loop planar mechanisms. The two approaches which look promising are the

- 1) complex number method

---

\*Numbers in parentheses refer to numbered references in the bibliography.



## 2) displacement matrix method

The complex number method arrives at the synthesis equations by making use of the complex vectors and eliminating unwanted parameters. The disadvantage with this method is that for every mechanism, the synthesis equations are to be obtained from scratch. Its advantage over the displacement matrix method is that it involves less number of unknowns when compared to the synthesis equations obtained by displacement matrix method. The greatest advantage which the displacement method offers is that the synthesis equations can be obtained with ease and therefore gives the opportunity of computerization of the method. Kohli and Soni (3), and Kim, Hamid and Soni (4) used the displacement matrix method successfully in synthesizing the seven link and the six link mechanisms. Hamid and Soni (7) solved the eight link mechanism for a variety of motion programs using the displacement matrix method.

The work that was undertaken in this thesis was with the aim to computerize the whole system of mechanisms up to and including all the eight link mechanisms so that synthesis equations can be obtained directly as the computer output. The program is written in PL/I and FORMAC to interpret the mathematical terms.

Chapter II explains the use of the displacement matrix. In Chapter III, the definitions and notations of some of the terms related to kinematics are given. Chapter IV explains the data required as input to the program. Selection of the coupler link is explained in Chapter V, and Chapter VI explains how the synthesis equations are obtained.

Listing of the computer program is given in the Appendix.

## CHAPTER II

### DERIVATION AND EXPLANATION OF DISPLACEMENT AND ROTATION MATRICES

In this chapter, the displacement and rotation matrices are derived using the elementary principles of algebra and trigonometry. An explanation is also given as to their use and significance. Suh (2) was the person who first formulated and used these matrices to obtain design equations for planar mechanisms. He later extended his work to include the three dimensional mechanisms, also.

Referring to Figure 1, let  $\Sigma_1$  and  $\Sigma_N$  represent two different positions of any link, say  $\Sigma$ , of a mechanism when the mechanism has moved from its 1<sup>st</sup> position to any other position, say N.

$A_1$  and  $B_1$  are any two points on the link  $\Sigma$  when the mechanism was in its 1<sup>st</sup> position, and  $A_N$  and  $B_N$  are the same two corresponding points when the mechanism has moved to its new position, N.

Let  $(XA_1, YA_1)$  be the coordinates of point  $A_1$

$(XB_1, YB_1)$  be the coordinates of point  $B_1$

$(XAN, YAN)$  be the coordinates of point  $A_N$

$(XBN, YBN)$  be the coordinates of point  $B_N$

$\alpha$  is the angle which the straight line  $A_1B_1$  makes with the horizontal.

$\theta$  is the angle by which the link  $\Sigma$  has rotated in moving from its 1<sup>st</sup> position to its new position, N.

From Figure 1(a),

$$\cos(\alpha) = \frac{A_1 D}{A_1 B} = \frac{XB_1 - XA_1}{A_1 B_1} \quad (1)$$

$$\sin(\alpha) = \frac{B_1 C}{A_1 B_1} = \frac{YB_1 - YA_1}{A_1 B_1} \quad (2)$$

and from Figure 1 (b)

$$\cos(\alpha+\theta) = \frac{A_N D}{A_N B_N} = \frac{XBN - XAN}{A_N B_N} \quad (3)$$

By trigonometry,  $\cos(\alpha+\theta)$  can be expanded to get

$$\cos(\alpha+\theta) = \cos(\alpha) \cdot \cos(\theta) - \sin(\alpha) \cdot \sin(\theta) \quad (4)$$

Substitute for  $\cos(\alpha)$ ,  $\sin(\alpha)$  AND  $\cos(\alpha+\theta)$  using expressions (1), (2), and (3) in expression (4) to get

$$\frac{XBN - XAN}{A_N B_N} = \frac{(XB_1 - XA_1)}{A_1 B_1} \cdot \cos(\theta) - \frac{(YB_1 - YA_1)}{A_1 B_1} \cdot \sin(\theta) \quad (5)$$

$A_N B_N$  can be cancelled out with  $A_1 B_1$  as both of them represent the same distance between the two points A and B. After cancelling out  $A_N B_N$  with  $A_1 B_1$  and rearranging to get XBN, the expression becomes

$$XBN = XB_1 \cdot \cos(\theta) - YB_1 \cdot \sin(\theta) + [XAN - XA_1 \cdot \cos(\theta) + YA_1 \cdot \sin(\theta)] \quad (6)$$

Similarly, the expression for YBN can be obtained by expanding  $\sin(\alpha+\theta)$  as

$$YBN = XB_1 \cdot \sin(\theta) + YB_1 \cdot \cos(\theta) + [YAN - XA_1 \cdot \sin(\theta) - YA_1 \cdot \cos(\theta)] \quad (7)$$

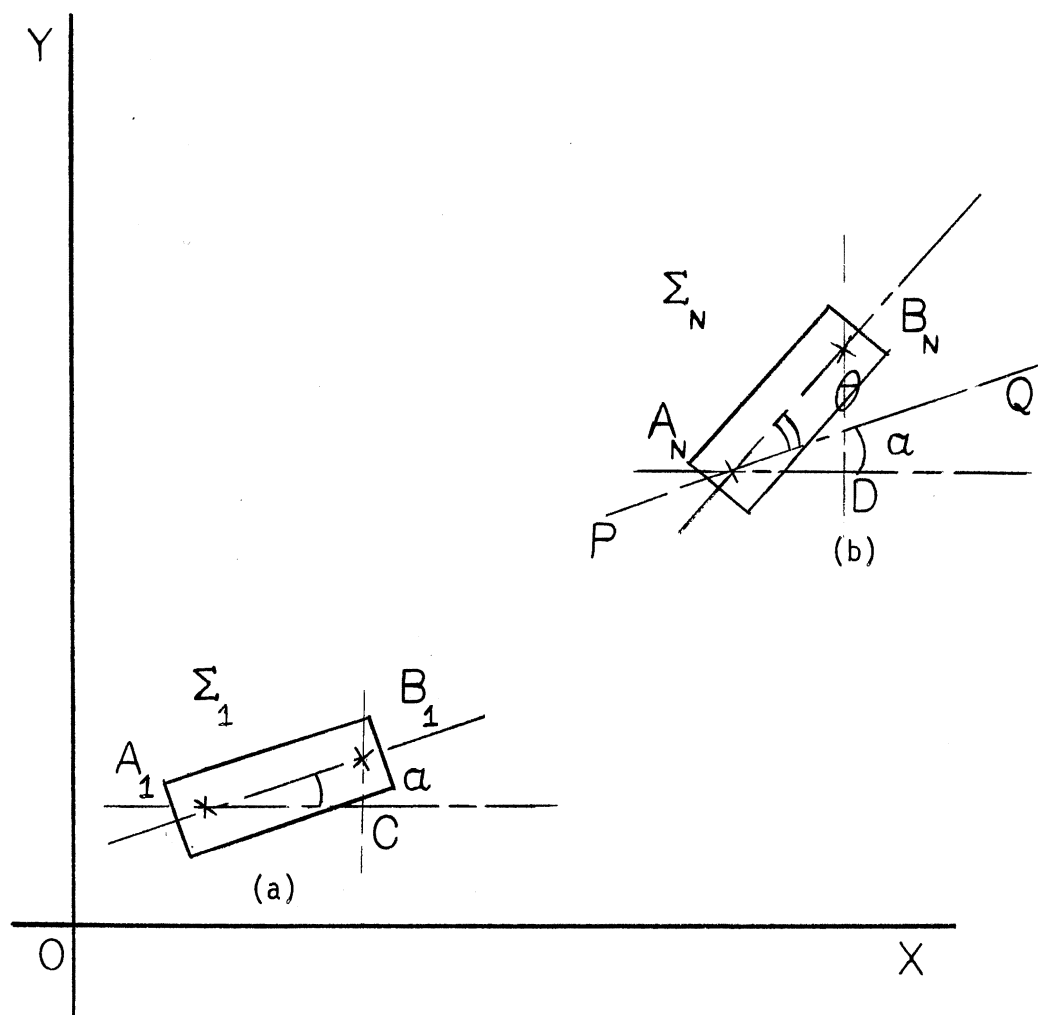


Figure 1. Displacement and Rotation of a Planar Link

Expressions (6) and (7) can be expressed in matrix form to get

$$\begin{bmatrix} X_{BN} \\ Y_{BN} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & X_{AN} - X_{A1} \cdot \cos(\theta) + Y_{A1} \cdot \sin(\theta) \\ \sin(\theta) & \cos(\theta) & Y_{AN} - X_{A1} \cdot \sin(\theta) - Y_{A1} \cdot \cos(\theta) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{B1} \\ Y_{B1} \\ 1 \end{bmatrix}$$

$$\text{Let } \begin{bmatrix} D \end{bmatrix}_{IN} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & X_{AN} - X_{A1} \cdot \cos(\theta) + Y_{A1} \cdot \sin(\theta) \\ \sin(\theta) & \cos(\theta) & Y_{AN} - X_{A1} \cdot \sin(\theta) - Y_{A1} \cdot \cos(\theta) \\ 0 & 0 & 1 \end{bmatrix}$$

The above matrix  $D_{IN}$  is defined as the displacement matrix. It may be noted that it is a function of the rotation angle  $\theta$  and the co-ordinates of point A in two different positions.

Thus, if the angle  $\theta$  and the coordinates of point A were known in both the positions 1 and N, then the displacement matrix  $D_{IN}$  is completely known.

If the coordinates of point B in its first position were known, then its coordinates in the  $N^{\text{th}}$  position can be determined using the displacement matrix  $D_{IN}$  from the relation

$$\begin{bmatrix} X_{BN} \\ Y_{BN} \\ 1 \end{bmatrix} = \begin{bmatrix} D \end{bmatrix}_{IN} \begin{bmatrix} X_{B1} \\ Y_{B1} \\ 1 \end{bmatrix}$$

It is important to note here that by expressing the coordinates of point B in its  $N^{th}$  position in terms of its coordinates in the  $I^{st}$  position using the displacement matrix  $D_{IN}$  which is a function of the coordinates of point A in two different positions, what is indirectly being said is that these two points, A and B, lie on the same plane. This point must be clearly understood in order to understand the work reported in this thesis. Thus, generalizing if there are  $m$  points  $B_1, B_2, B_3, \dots, B_m$  and  $D_{IN}$ , the displacement matrix is a function of the coordinates of some point A and the angle  $\theta$  and if the coordinates of all these  $m$  points  $B_1, B_2, \dots, B_m$  in their  $N^{th}$  position is expressed in terms of their first position using this displacement matrix  $D_{IN}$  then it is to be understood that all these  $m$  points  $B_1, B_2, \dots, B_m$  and the point A lie on the same plane.

In Figure 2, the link has rotated only by an angle  $\theta$  about the point A in moving from position  $\Sigma_1$  to position  $\Sigma_N$ . As point A has not moved at all, its coordinates in position  $\Sigma_N$  is the same as that in position  $\Sigma_1$ . That is,  $X_{AN} = X_{A1}$  and  $Y_{AN} = Y_{A1}$ . Substituting for  $X_{AN}$  and  $Y_{AN}$  using these relations in the displacement matrix, the rotation matrix is obtained.

$$[R]_{IN} = \begin{bmatrix} \cos(\theta) - \sin(\theta) & X_{A1} (1 - \cos(\theta)) + Y_{A1} \sin(\theta) \\ \sin(\theta) & \cos(\theta) & Y_{A1} (1 - \cos(\theta)) - X_{A1} \sin(\theta) \\ 0 & 0 & 1 \end{bmatrix}$$

Thus, in Figure 2 the coordinates of point B in position  $\Sigma_N$  can be obtained by using the relation

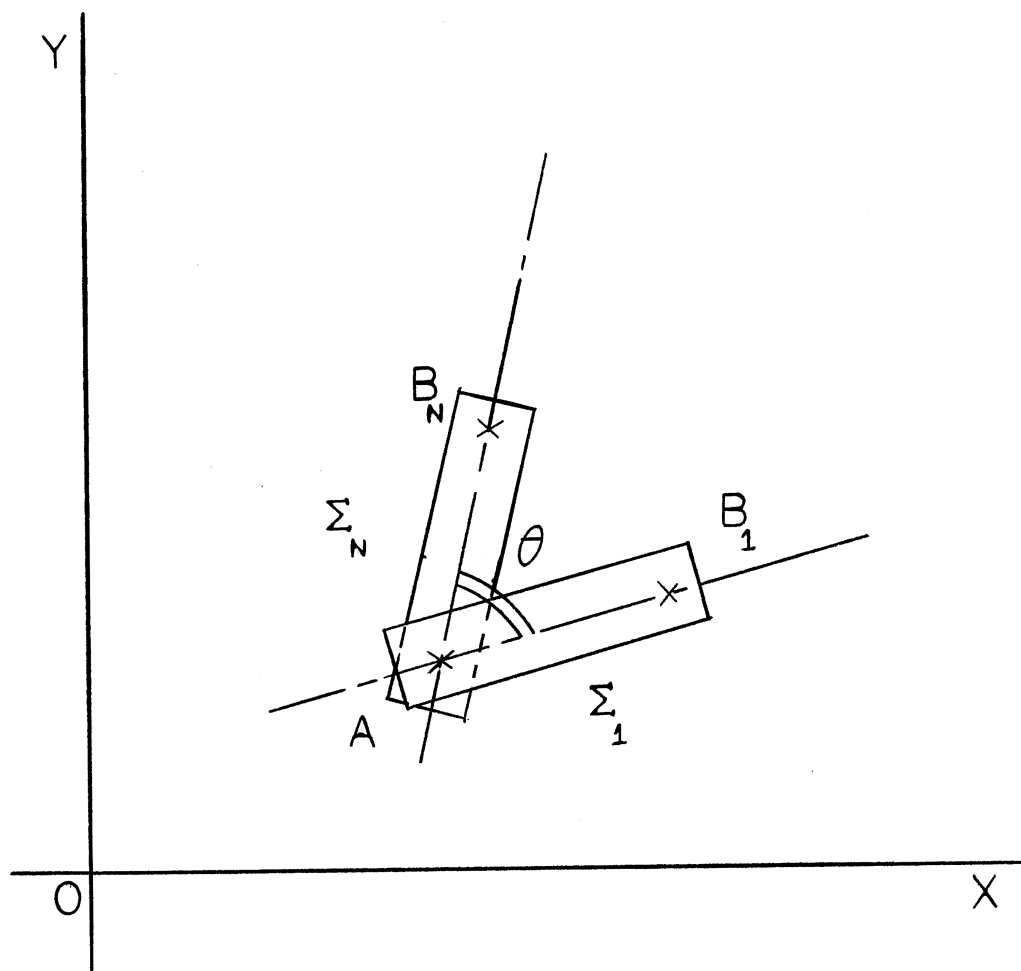


Figure 2. Rotation of a Planar Link

$$\begin{bmatrix} X_{BN} \\ Y_{BN} \\ 1 \end{bmatrix} = \begin{bmatrix} R \end{bmatrix}_{IN} \begin{bmatrix} X_{B1} \\ Y_{B1} \\ 1 \end{bmatrix}$$

in terms of its coordinates in first position. Following the same reasoning which was used for displacement matrix, it can also be argued on similar lines that the use of the rotation matrix between two or more points implies that they all lie on the same plane.



## CHAPTER III

### DEFINITION OF NOTATIONS AND TERMS FOR THE SYMBOLIC COMPUTER PROGRAM

In this chapter, a brief explanation of the different terms which are used regarding a mechanism will be given.

A kinematic chain is a geometric configuration built by connecting kinematic links using kinematic pairs. It may have one or more closed loops.

A kinematic link is a rigid massless member and is classified according to the number of joints which it can accept. A binary link is defined as a kinematic link which can accept two joints; a ternary link, three joints, and a K-nary link, K joints. A kinematic pair consists of two contacting elements and is used to connect two kinematic links.

A linkage or a mechanism is obtained by fixing one of the kinematic links of a closed kinematic chain with the ground. The link which is fixed with the ground is called the ground link, and the links which are connected to the ground link are called the ground adjacent links.

In the kinematic chain, each link must be free to move relative to the other links connected to it. In a basic planar kinematic chain, all the links are connected by revolute pairs, and the movement of all the links is in one or different parallel planes.

A kinematic mechanism obtained by fixing one of the links of a

kinematic chain may have one or more degrees of freedom. The degrees of freedom of a mechanism depend upon the total number of links and kinematic pairs present in it. The degrees of freedom of a mechanism can be determined by using Gruebler's mobility criterion, and for planar mechanisms, the degrees of freedom are given by  $F = 3(N - 1) - 2P_1$  where

$F$  = degrees of freedom of the mechanism

$N$  = total number of links the mechanism has

$P_1$  = total number of revolute pairs present.

In a single degree of freedom mechanism, the motions of all the links are constrained; that is, no link is free to move independent of the other links. In a multiple degree of freedom mechanism, there must be as many inputs as there are degrees of freedom to obtain constrained motion between the different links of the mechanism.

A dyad is defined to be the configuration of two binary links connected by a revolute pair. It will be shown below that by adding a dyad to or subtracting it from a single degree of freedom mechanism, the degree of freedom of the mechanism is not changed.

Let  $X$  number of links and  $Y$  number of revolute pairs be added to a mechanism which already has  $N$  number of links and  $P_1$  number of revolute pairs. The expression for the degree of freedom becomes

$$F = 3(N + X - 1) - 2(P_1 + Y)$$

Rearrange the above expression to get

$$F = 3(N - 1) - 2P_1 + 3X - 2Y$$

Thus, in the above expression, if  $X$  is equal to two and  $Y$  three,

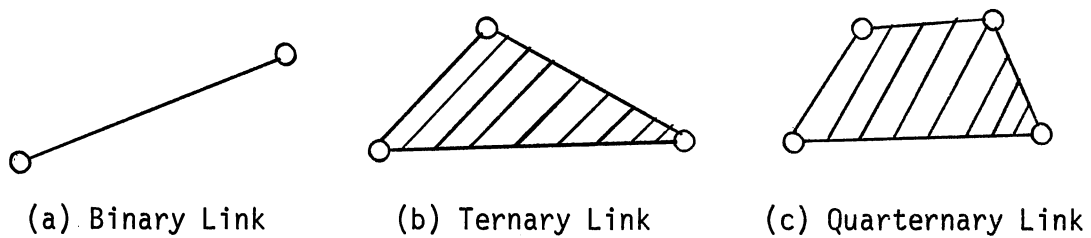


Figure 3. Kinematic Links

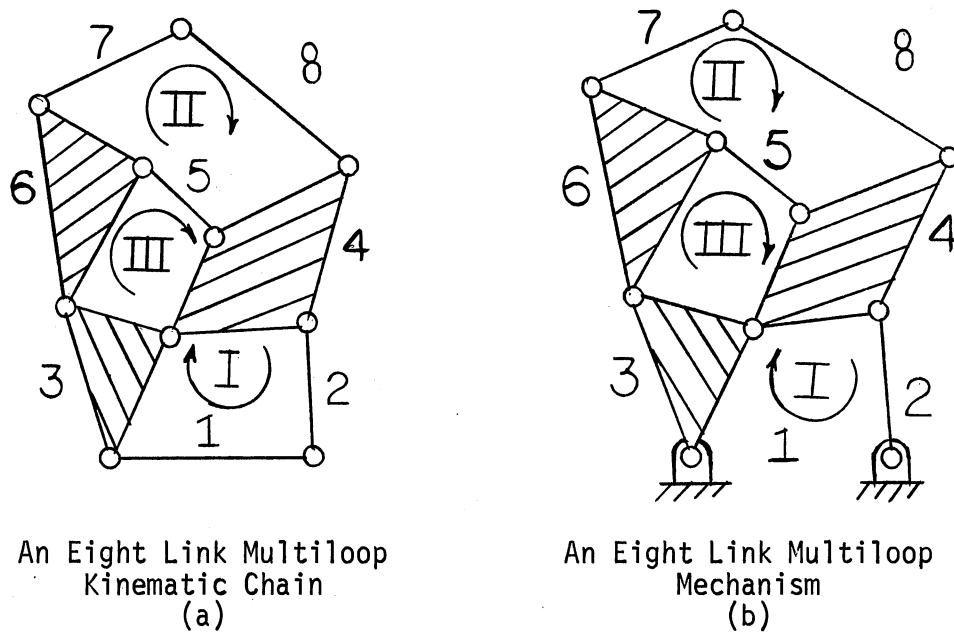


Figure 4. An Eight Link Kinematic Chain and an Eight Link Mechanism

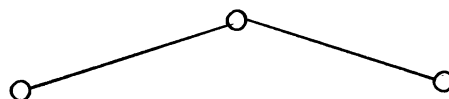


Figure 5. A Dyad

the expression for the degree of freedom of the mechanism will not change and this proves the statement made above. Figure 6 shows how an eight link mechanism reduces to a six link mechanism by removing the dyad from the eight link mechanism. This point will be made use of in selecting the coupler link of the mechanism explained in Chapter V.

This provides a sufficient background in understanding the remaining chapters of this thesis.

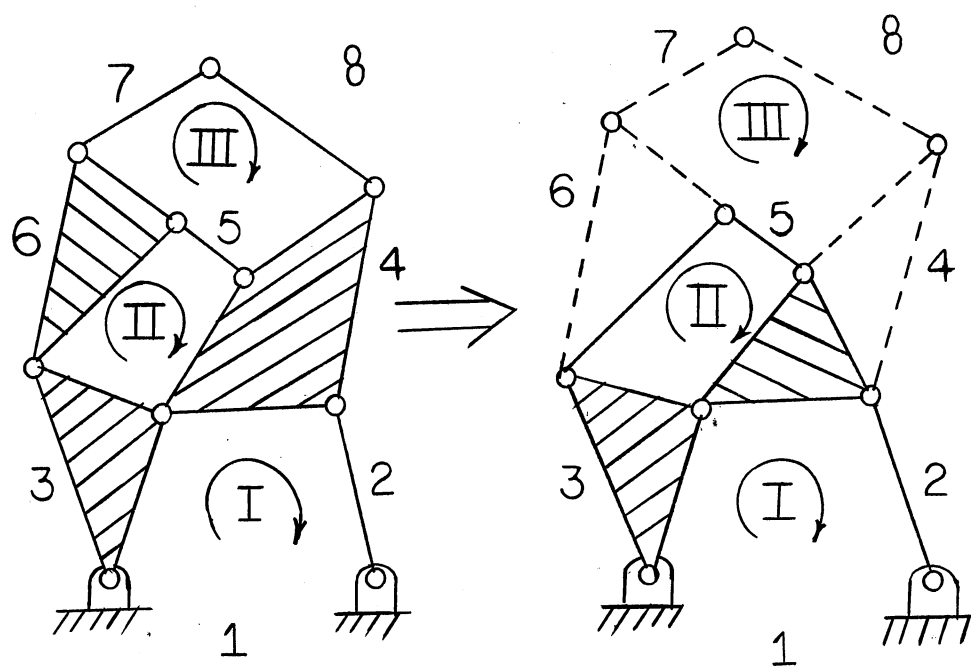


Figure 6. Removal of a Dyad

## CHAPTER IV

### INPUT INFORMATION REQUIRED FOR THE COMPUTER PROGRAM

In this section, the scheme that was adopted for describing a mechanism to a high speed digital computer will be explained in detail.

To begin with, the different links of a mechanism must be numbered so that they can be identified later, by referring to these numbers. The numbering of the links is completely independent of the computer program (a listing of which is given in the Appendix). But when once the links are numbered, the names of the different pivot points of the mechanism are fixed by the computer program. How this is done is explained below.

The information regarding the connection between different links of a mechanism can very well be represented by a two-dimensional array. In the computer program, this array is named MECH. If, for example, say link numbers I and J of a mechanism are connected to each other by a revolute pair, then the locations (I, J) and (J, I) in the array MECH will have a value of 1 (ONE), and if there were no connection between the links, then a 0 (ZERO) would occupy the above mentioned locations.

Thus, the complete array MECH for mechanism number two shown in Figure 7 will take the following form:

MECH(I,J) =

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	1	0	1
2	0	0	0	0	0	0	1	1
3	0	0	0	0	0	0	1	1
4	0	0	0	0	1	1	0	0
5	0	0	0	1	0	0	1	0
6	1	0	0	1	0	0	1	0
7	0	1	1	0	1	1	0	0
8	1	1	1	0	0	0	0	0

It may be noted that MECH is a square array symmetric about the principal diagonal. Now consider the upper half portion of this array where the diagonal beginning at the element (1, 1) divides it into two parts. Counting the non-zero entries in this portion, it will be found that there are as many of them as there are number of pivot points in the mechanism. We now modify the matrix MECH using the following rules. Note that the pivot points of the mechanism are identified by using the letters A, B, C, etc. Replace the non-zero entries in the matrix MECH by these letters representing the pivot points. Thus, for example, the connectivity between links 1 and 6, replace the non-zero entry for the element (1, 6) by the letter A. The modified matrix MECH will then

take the following form:

MECH(I,J) =

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	A	0	B
2	0	0	0	0	0	0	C	D
3	0	0	0	0	0	0	E	F
4	0	0	0	0	G	H	0	0
5	0	0	0	G	0	0	J	0
6	A	0	0	H	0	0	K	0
7	0	C	E	0	J	K	0	0
8	B	D	F	0	0	0	0	0

Note that the letter I is not used in writing this modified matrix MECH. This is simply to avoid confusion between the letter I and the number 1. Since an eight link mechanism has ten joints, the letter symbols of these ten joints will therefore be A, B, C, D, E, F, G, H, J, and K. The letter P is used for the coupler point.

Consider the two mechanisms shown in Figure 7. By examining these mechanisms we note the following distinguishing points:

- 1) They do not have the same number of links



- 2) All the links in either of them are not of the same type.
- 3) The total number of loops in each of them are not the same.

Apart from the above observations, another important point which must also be noted is about the connection between the different links. For example, link number 1 of mechanism number 2 in Figure 7 is not connected to link number 7, and such information regarding the mechanism is important and must be carefully noted.

A mechanism may have binary, ternary, quaternary, and other types of links present in it. This information is made known to the computer by the single dimensional array named LINK. Thus, if  $\text{LINK}(I) = K$ , then this will mean that link number I of the mechanism is a K-nary link.

The other information required by the computer is regarding the loops of the mechanism, and must be carefully prepared. There are a few rules which must be following in preparing these data. These rules are:

- 1) All the different loops must follow the same path, either clockwise or counterclockwise.

- 2) The links which are encountered in traversing a loop must be noted down in a sequential order. No link is to be repeated or skipped.

- 3) The first loop must have the ground link as its first element.

- 4) If there is more than one loop having ground link as one of its elements, then the following rule should be adopted in naming the first loop:

- a) If all the loops are following the clockwise path, then the lefthand-most loop will be loop number 1.

- b) If all the loops are following the counterclockwise path,

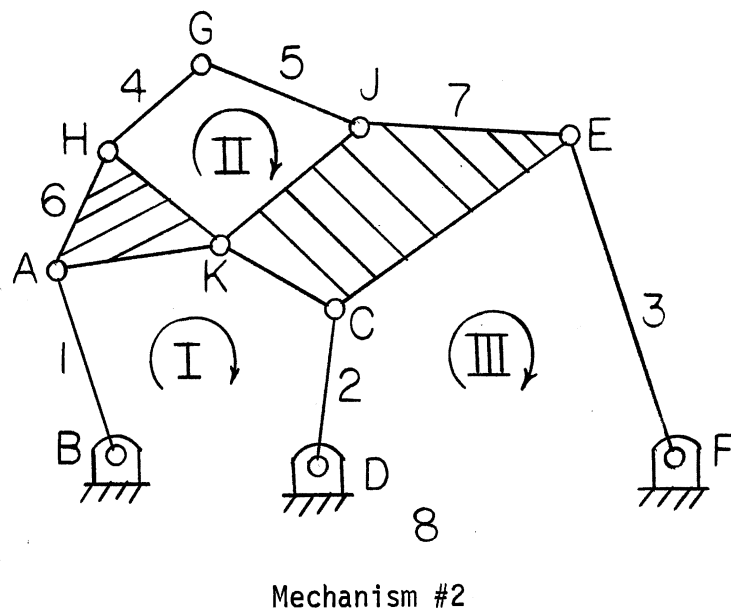
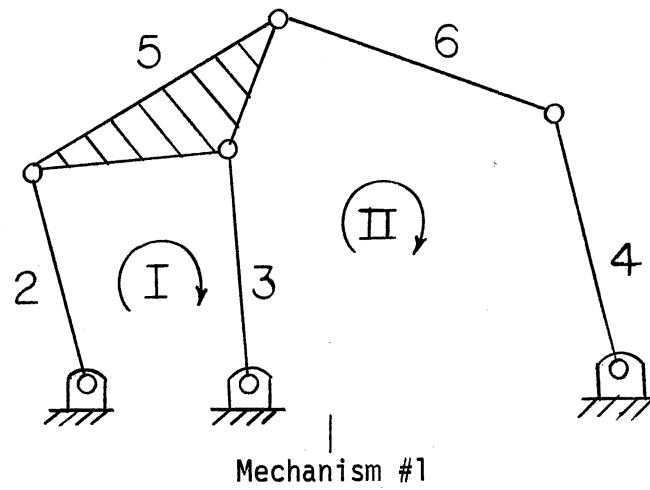


Figure 7. Comparison of Mechanisms

then the righthand-most loop will be loop number 1.

The numbering of other loops is immaterial and may be done in any manner.

The input data required for the computer program is explained below and should be given in the order shown:

- 1) total number of links in the mechanism
- 2) the number of the ground link
- 3) total number of loops present in the mechanism
- 4) total number of elements in each of the loops beginning with the first loop first and following it with other loops in an increasing order
- 5) link numbers encountered in each of the loops beginning with loop number 1 and following it with other loops in an increasing order
- 6) numbers indicating link types for all eight links in a sequential order; that is, begin with link number 1 and state its type, then go to link number 2 and state its type, and end with link number 8 and state its type
- 7) the data which are obtained from the upper half portion of the non-modified matrix MECH.

This completes the input data for the computer program. Listing of the data which was prepared for mechanism number 2 of Figure 7 are presented on the next page. Until item number 6) above the data are punched on the data cards with a blank between each value. The data for the item number 7) are to be given as shown in the listing. Note that a comma appears between each value and the input data must end with a semi-colon.

LISTING OF THE INPUT DATA FOR THE  
EIGHT LINK MECHANISM OF FIGURE 7

8 8 3 5 4 4 8 1 6 7 2 7 6 4 5 2 7 3 8 2 2 2 2 2 3 4 3 MECH(1,6)=1,MECH(1,8)=1,ME

CH(2,7)=1,MECH(2,8)=1,MECH(3,7)=1,MECH(3,8)=1,MECH(4,5)=1,MECH(4,6)=1,MECH(5,7)=

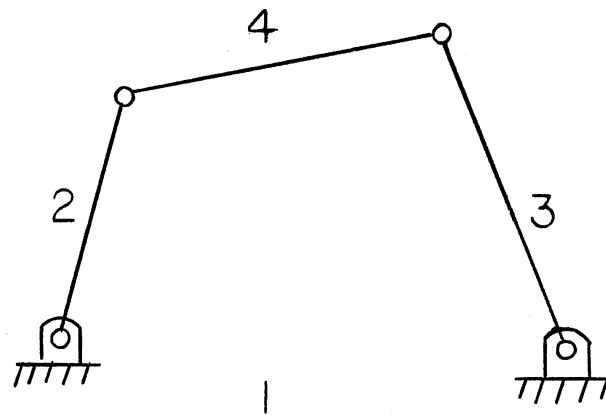
1,MECH(6,7)=1;

## CHAPTER V

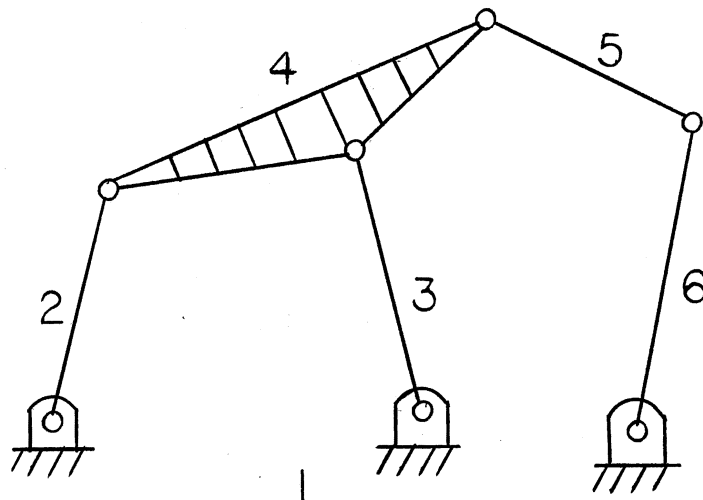
### SELECTION OF COUPLER LINK

After a mechanism is selected, it becomes important to decide upon a link which will be used as the coupler link. "Coupler link" is a typical name given to a particular link of the mechanism which will actually do the job of generating a point path curve or guiding a rigid body through its different precision positions.

In a simple four bar mechanism, the link which connects the two ground adjacent links is selected as the coupler link. It is known that a four bar mechanism is capable of generating a curve having a maximum of nine precision points. In Figure 8, a four bar mechanism is made into a six bar mechanism by adding a dyad. The coupler link of the simple four bar mechanism is link number 4. The question now is, by having added a dyad to the four bar mechanism, will the coupler link remain the same, or is some other link to be chosen as the coupler link of this six bar mechanism? For a moment, let the original link number 4 be selected as the coupler link of this six bar mechanism. It is known that a six bar mechanism can be synthesized for a maximum number of 15 precision positions. With coupler link as link number 4, let the above mentioned six bar mechanism be synthesized for its maximum number of precision positions. Now, having synthesized the mechanism for its maximum number of precision positions, remove the dyad to get back the original four bar mechanism. With the above procedure, actually a four



(a) A Four Link Mechanism



(b) A Six Link Mechanism

Figure 8. Adding a Dyad to a Four Link Mechanism

bar mechanism has been synthesized for 15 precision positions. This is not possible, however, since a four bar mechanism can be synthesized for a maximum of nine precision positions only. If the above discussion were to hold, then by similar reasoning it can be argued that a four bar mechanism can be synthesized for an infinite number of precision positions by adding infinity number of dyads to the coupler plane of the four bar mechanism as shown in Figure 9. This is obviously not true, and therefore link number 4 cannot be selected as the coupler link of the six bar mechanism. The only other possibility is for link number 5 to be selected as the coupler link.

From the above discussion, the following rules can be made in selecting a coupler link:

- 1) If there is a dyad present in the mechanism (not taking into account the ground link), then one of the binary links of the dyad must be selected as the coupler link of the mechanism.

- 2) If there is more than one dyad present in the mechanism, then the mechanism is not a true representation of its class. For example, an eight link mechanism having two dyads can be reduced to a six link mechanism by removing one of the dyads. Such mechanisms must not be considered as a true eight link mechanism.

- 3) The coupler link must be so selected that it is farthest away from the ground link. If there happen to be two such links, then the one having a less number of revolute pairs is to be selected as the coupler link. The simple reason behind this rule is the argument that a binary link will be more free to move when compared to a ternary link, and so is the  $(K-1)$ -nary link when compared to a  $K$ -nary link.

An example mechanism will now be selected and a complete step-by-



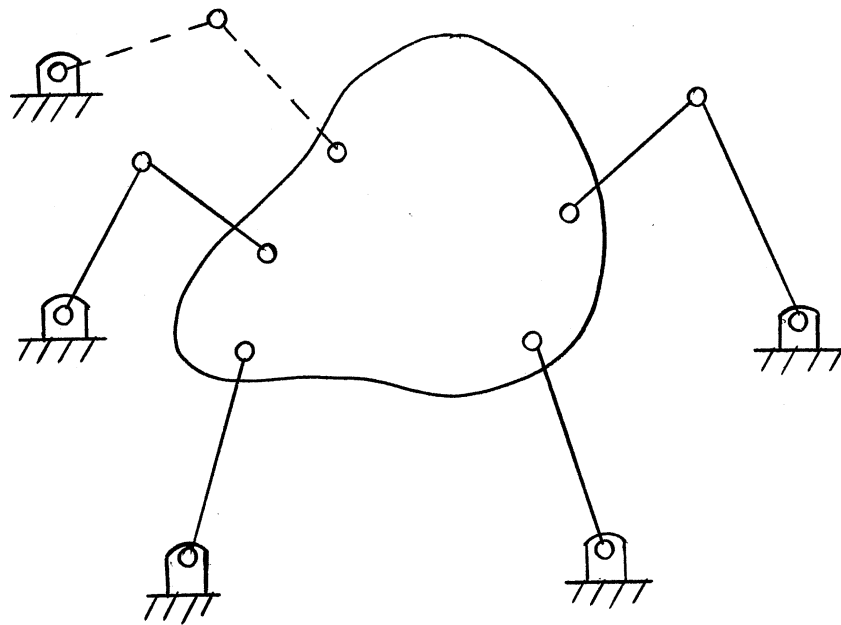


Figure 9. Dyads Added to a Four Link Mechanism

step procedure will be explained as to how the computer program goes about in selecting the coupler link of the mechanism. The example mechanism is shown in Figure 10.

Following the clockwise rule, let the loops of the mechanisms be numbered as shown in Figure 10. Therefore, the input data regarding the loops will be:

LOOP (1, \*) = 8, 1, 6, 7, 2

LOOP (2, \*) = 4, 5, 7, 6

LOOP (3, \*) = 7, 3, 8, 2

If the mechanism is other than the four link mechanism, then procedure No. 1 is called, or else the coupler link is selected to be that link which occupies the third location in LOOP (1, \*).

After procedure No. 1 is called, the following steps are executed before the control returns to the main program.

STEP 1: Procedure No. 2 is called; it determines the total number of dyads present in the mechanism. If there is more than one dyad present, then a message is printed out saying that there is more than one dyad present in the mechanism, and then the program stops.

STEP 2: If there is only one dyad present in the mechanism, procedure No. 3 is called; it checks to see if one of the binary links of this dyad is connected to the ground link and, if so, then the other link is selected as the coupler link. The control returns to procedure No. 1. If the coupler link is selected, then a check is made to see if the total number of links in the loop of which the dyad is a part, is four. If this is true, then this is a pseudo eight link mechanism, and a message regarding it will be printed before the program ends.

STEP 3: LOOP number 1 is stored in the first row of a two

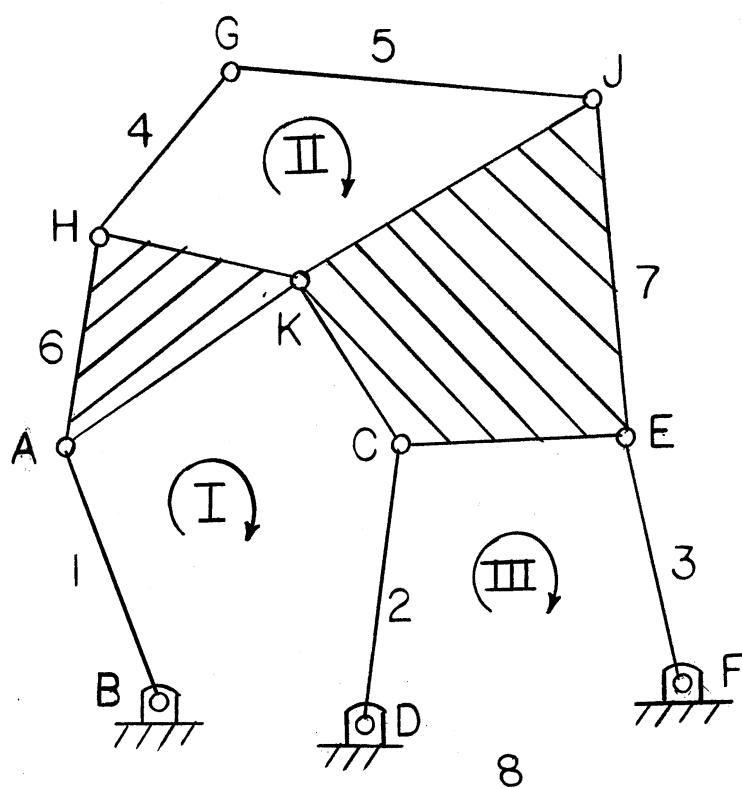


Figure 10. An Eight Link Mechanism

dimensional array named LUPCOM (short for loop combined). Thus, the first row of LUPCOM will be

$$\text{LUPCOM}(1, *) = 8, 1, 6, 7, 2.$$

Procedure No. 4 is now called; it selects a loop other than the first loop which will have at least two links in common when compared to  $\text{LUPCOM}(1, *)$ . The ground link is not considered in determining the common links when the above selection is being made. Thus, loop number 2 will be selected which has link numbers 6 and 7 in common with  $\text{LUPCOM}(1, *)$ . The locations of these two common links in  $\text{LOOP}(2, *)$  will be also noted. Thus, link number 6 occupies the fourth location and link number 7 the third in  $\text{LOOP}(2, *)$ . As link number 6 was first found to be the common link, a check will be made to see if its location is less than that of the other common link which is link number 7. If this is not found to be true, then  $\text{LOOP}(2, *)$  will be rearranged one or more number of times if necessary until the link number which was first found to be common occupies a location which is less than the location occupied by the other common link. The rearrangement of loops is done by calling procedure No. 5. The rearrange loop will look as shown below:

$\text{LOOP}(2, *) = 4, 5, 7, 6$ --ORIGINAL

$\text{LOOP}(2, *) = 5, 7, 6, 4$ --Rearranged once

$\text{LOOP}(2, *) = 7, 6, 4, 5$ --Rearranged twice

$\text{LOOP}(2, *) = 6, 4, 5, 7$ --Rearranged thrice

As the location of link number 6 is less than the location of link No. 7, procedure No. 5 returns control to procedure No. 1.

STEP 4: The rearranged loop is now ready to be combined with  $\text{LUPCOM}(1, *)$  and the resultant combined loop is stored in the next row

of LUPCOM. The combination is done as follows:

LUPCOM (1, \*) = 8, 1, 6, 7, 2

LOOP (2, \*) = 6, 4, 5, 7

All the links in LUPCOM (1, \*) up to and including the link which was first found to be in common is stored in the next row of LUPCOM. Thus, the first few elements of LUPCOM (2, \*) will be

LUPCOM (2, \*) = 8, 1, 6, \* \* \*

After storing these elements, the control is now transferred to LOOP (2, \*) at that link number which was first found to be common that is at the first location of LOOP (2, \*). The link number which is occupying the next location in LOOP (2, \*) is now stored as the new element in LUPCOM (2, \*). A check is now made to see whether this link is present in LUPCOM (1, \*) and, if so, the control is transferred back to LUPCOM (1, \*) at the location occupying this link number. If this were not true, a second check is made to see if this link is connected to the ground link. The combination is ended if this second check comes out to be true. As both these checks turn out to be false, the combination will continue until link No. 7 in LOOP (2, \*) is stored in LUPCOM (2, \*). Control is now transferred to LUPCOM (1, \*), and finally link No. 2, the last element of LUPCOM (1, \*) is stored in LUPCOM (2, \*). The combination of LUPCOM (1, \*) and LOOP (2, \*) will therefore look like

LUPCOM (2, \*) = 8, 1, 6, 4, 5, 7, 2.

STEP 5: If the total number of loops is greater than the number of the loop which was combined, then procedure No. 7 will be called which will update all the loops whose numbers are greater than the number of the loop which was combined. To make it more clear as to what has been said above, consider that LOOP (2, \*) has been combined

with LUPCOM (1, \*) and as the number of the combined loop which is 2 is less than the total number of loops which is 3, procedure No. 7 will be called. This procedure stores loop No. 3 in loop No. 2. This is what is meant by updating the loops. Therefore, LOOP (2, \*) will now become

$$\text{LOOP (2, *)} = 7, 3, 8, 2.$$

The same procedure is followed all over again, this time to find a loop which will have at least two links in common when compared with LUPCOM (2, \*) so that combination of LUPCOM (2, \*) with this loop can be made. This new combination is stored in the next row of the array LUPCOM. All the steps involved in the combination will not be explained again as they have already been explained in detail above. The combination of LUPCOM (2, \*) with the updated LOOP (2, \*) will look like

$$\text{LUPCOM (3, *)} = 8, 1, 6, 4, 5, 7, 3.$$

The total number of links in each of the combined loops is noted after each has been determined.

Thus, the total number of links in each of the combined loops is 5, 7, 7.

STEP 6: The coupler link is selected from that combined loop which has a maximum number of links in it. If there happens to be two or more such loops having the same number of links, then that loop whose number is greater than the others is chosen for coupler link selection. Thus, LUPCOM (3, \*) will be chosen from which a possible link will be selected as the coupler link. The total number of links in LUPCOM (3, \*) is 7, which is an odd number and this indicates that there are two links which must be considered for possible coupler link selection. Treating 7 as an integer variable and dividing it by 2 will

give 3 and adding 1 and 2 to this number gives 4 and 5. Thus, links occupying locations 4 and 5 in LUPCOM (3, \*) are to be tested. Now

$$\text{LUPCOM}(3, 4) = 4 \text{ and } \text{LUPCOM}(3, 5) = 5.$$

The two links, link numbers 4 and 5, will be checked to determine their type and the one having less number of pairs on it will be selected as the coupler link. If the total number of links in the combined loop (which has the maximum number of links in it) were even, then the link occupying the location given by half the total number of links plus one would be selected as the coupler link.

STEP 7: Finally, procedure No. 10 will be called when there is a dyad present in the mechanism to check if the coupler link which has been selected happens to be a binary link of this dyad. If this is not true, then that combined loop is searched out which has the dyad and the optimum binary link of this dyad will then be selected as the coupler link. After the selection is made, the control returns to the main program.

## CHAPTER VI

### OBTAINING SYNTHESIS EQUATIONS

After having selected the coupler links, the next step is to obtain the synthesis equations for the mechanism. There are many different ways of writing the synthesis equations for the same mechanism. It therefore becomes necessary to generalize such a process as much as possible for computerization purposes. All the 71 eight link, the 5 six link, and the single four link mechanisms were studied in detail and it was found that all of these mechanisms excepting two can be divided into three groups. These two were eight link mechanisms having only binary and ternary links. Let these two mechanisms make the fourth group. Two separate procedures, numbered 18 and 19, were written for these two mechanisms of group four. Fortunately, these two mechanisms have unique features by which they can be detected from the 35 eight link mechanisms having only binary and ternary links.

The synthesis equations are so obtained that there is always an unknown angle associated with the coupler link. The advantage in obtaining the synthesis equations involving the rotation angle of the coupler link is for the fact that the same equations can be used either for point-path generation or for rigid body guidance type of problem.

Before proceeding further, let the following notations be adopted which will facilitate in understanding the process of obtaining the synthesis equations in a concise manner.



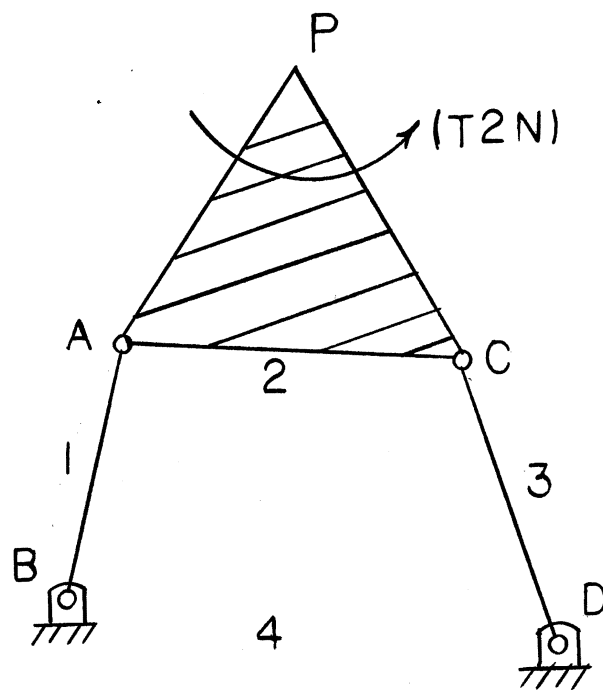


Figure 11. A Four Link Mechanism

Refer to the simple four bar mechanism shown in Figure 11. In order to obtain the synthesis equations for this mechanism, the following procedure is adopted: The coordinates of points A and C in their  $n^{\text{th}}$  position are expressed in terms of their first position, using a displacement matrix which is a function of the coordinates of a point P in two different positions, position 1 and position N and the rotation angle T2N of the link number 2 on which the points A, C, and P are located.

Let the above statement be represented mathematically as

$$A_N = f_1(P_1, P_N, T2N, A_1) \quad (1)$$

$$C_N = f_2(P_1, P_N, T2N, C_1) \quad (2)$$

Expression (1) conveys the meaning that the coordinates of pivot point A in its  $n^{\text{th}}$  position are being expressed in terms of their coordinates in first position using a displacement matrix which is a function of the coordinates of point P and the rotation angle T2N. This would generally have been represented as

$$\begin{bmatrix} X_{AN} \\ Y_{AN} \\ 1 \end{bmatrix} = \begin{bmatrix} D \end{bmatrix}_{1N} \begin{bmatrix} X_{A1} \\ Y_{A1} \\ 1 \end{bmatrix}$$

Going back to the mechanism shown in Figure 11, it is obvious that the link lengths AB and CD will remain the same regardless of the position which the mechanism will occupy. This gives rise to two constraining equations which are

$$(XAN - XB1)^2 + (YAN - YB1)^2 - (XA1 - XB1)^2 - (YA1 - YB1)^2 = 0$$

and

$$(XCN - XD1)^2 + (YCN - YD1)^2 - (XC1 - XD1)^2 - (YC1 - YD1)^2 = 0$$

For N number of precision positions, each of these two equations can be written (N-1) times. The total number of unknowns involved are the eight coordinates of the four pivot points A, B, C, and D and (N-1) unknown angles T2N of the coupler plane. Thus, for the system of equations to be consistent, there must be as many unknowns as there are numbers of equations. Equating these two

$$2(N-1) = 8 + (N-1)$$

the value of N can be determined. This gives  $N = 9$ , the maximum number of precision positions for which a four bar can be synthesized for point path generation. It is to be noted here that the number of constraining conditions is one more than the number of angles assumed to be unknowns in the equations. Thus, for any mechanism, the constraining conditions and the unknown angles associated with different links must be so selected that the total number of constraining conditions is always greater by one than the total number of unknown angles assumed.

The computer program has twenty-four different procedures or sub-routines, and each of them does a specific job when invoked by a call statement. The main program making use of the input data detects the group to which the mechanism belongs and it then transfers control to that portion of the program. These different portions of the main program will call the required procedures which results in the printout of

the synthesis equations. These required procedures are procedure Nos. 11 through 19. Each of these procedures will be explained briefly.

Procedure No. 11. This procedure when called will express all of the coordinates of the pivot points on the coupler link in terms of the coordinates of the coupler point P using a displacement matrix. Next, it searches for the different links connected to the coupler link and expresses the coordinates of the pivot points on those links which are not binary, in terms of the coordinates of the pivot point through which this link was connected to the previous link. An exception is that the coordinates of the pivot point which is on the ground link will not be considered.

Referring to Figure 12, the coordinates of pivot points K and D will be expressed in terms of the coordinates of the coupler point P, the coordinates of the pivot points, A, C, and B, will be expressed in terms of the coordinates of the pivot point D and finally the coordinates of the pivot point J will be expressed in terms of the coordinates of pivot point A.

Procedure No. 12. This procedure does a different kind of job. A typical portion of the mechanism for which this procedure is used is shown in Figure 13. When this procedure is called, it scans the mechanism to see if such portions as shown in Figure 13 are present in the mechanism. When such a construction is found, the following takes place.

The coordinates of the pivot points B and C (Figure 13) are expressed in terms of the coordinates of the pivot point A using a rotation matrix. Next, the coordinates of the pivot points E and D and F, G, and H are expressed, respectively, in terms of the coordinates of

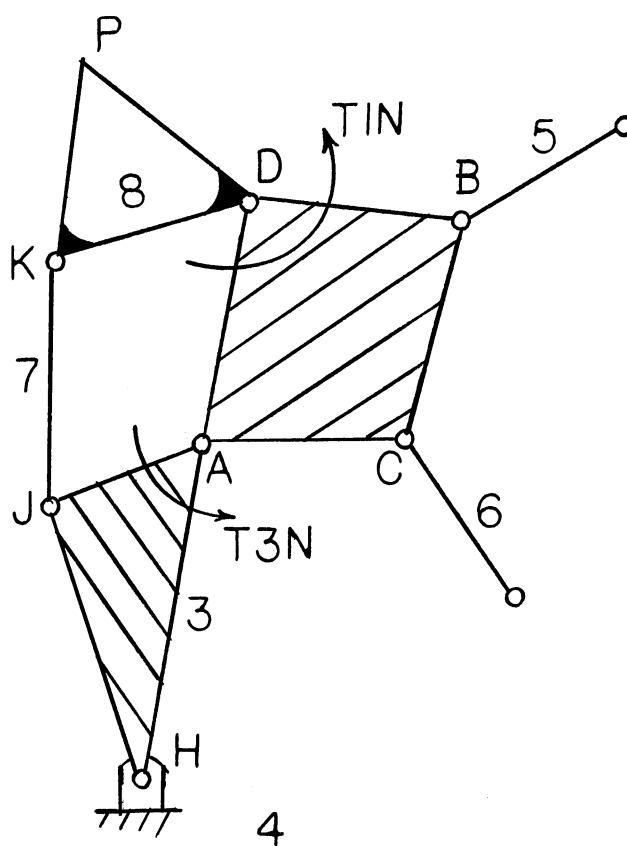


Figure 12. Figure for Explaining  
Procedure No. 11

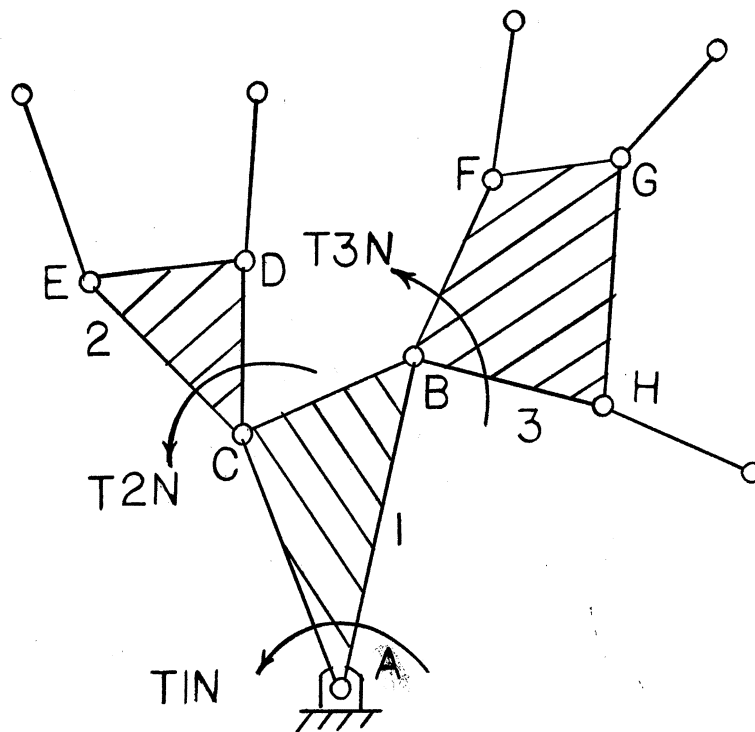


Figure 13. Figure for Explaining Procedure  
No. 12

the pivot points C and B. Either link number 1 or link number 3 or both of them can be binary links for procedure No. 12 to still be applicable.

Procedure No. 13. This procedure prints out constraining equations for any binary link encountered in the mechanism. The ground link and the coupler link are exceptions if they happen to be binary links.

Procedure No. 14. This procedure prints out constraining equations for the ground-adjacent links when portions as shown in Figure 12 are present in the mechanism. The sides JH and AH for the ground-adjacent link number 3 will form the constraining equations.

Procedure No. 15. This procedure will check the mechanism to see if it is a group-2 mechanism. A group-2 mechanism is shown in Figure 14. A mechanism is detected to be a group-2 mechanism when the following conditions are met:

- 1) It must have a total of eight links with no quaternary link in it.
- 2) A ground-adjacent link must be connected only by ternary links.
- 3) These ternary links must further not be connected by only binary links. A coupler link if it happens to be a binary link is not considered as a binary link connection.

If all three conditions are met, then the mechanism is classified as a group-2 mechanism. The three sides of such a ground-adjacent link forms three constraining equations.

Procedure No. 16. This procedure is called to check if a group-3 mechanism is given. A mechanism of this group is shown in Figure 15. It is detected from the fact that a ternary link is connected to the

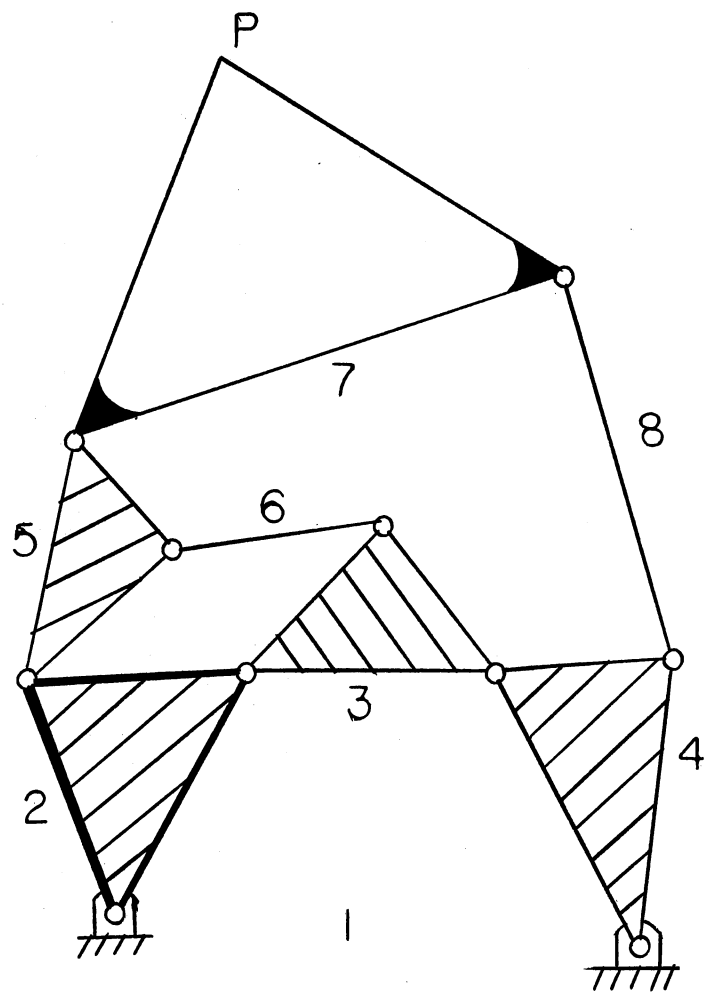


Figure 14. Group-2 Mechanism



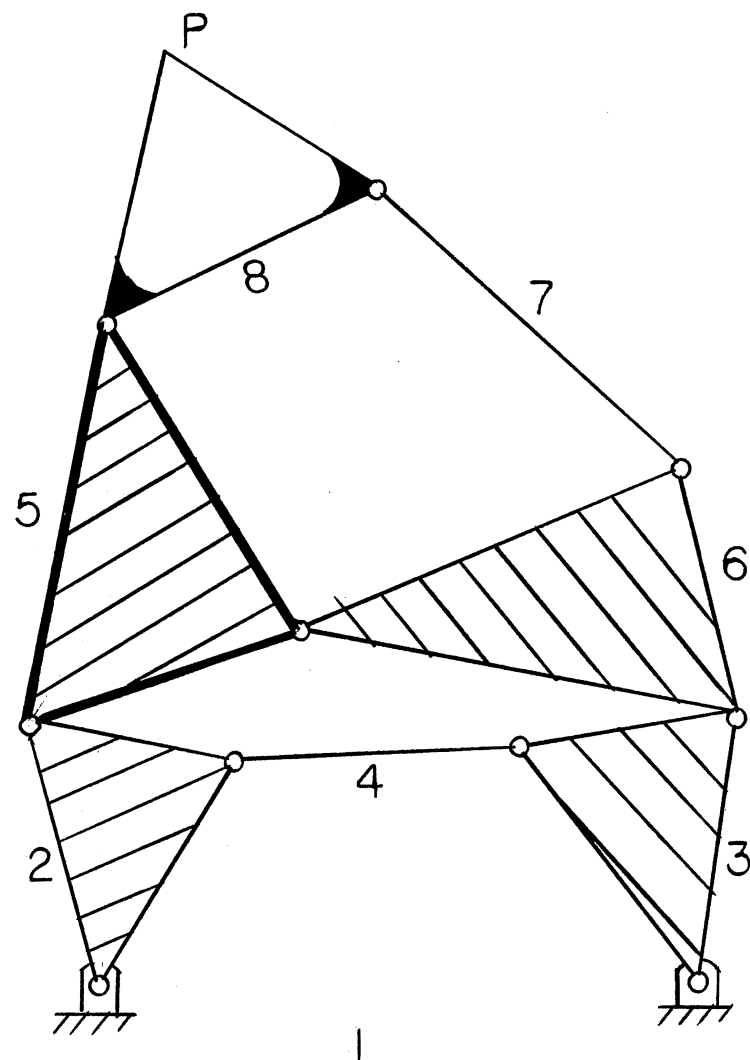


Figure 15. Group-3 Mechanism

coupler link and again in turn is connected by only ternary links. The three sides of this ternary link connected to the coupler link will form three constraining equations.

Procedure No. 17. This procedure prints out the constraining equations for the necessary ternary link either for a group-2 or group-3 mechanism.

Procedure No. 18 is called when mechanism #1 of group-4 is given.

Procedure No. 19 is called when mechanism #2 of group-4 is given.

An example mechanism of group-1 shown in Figure 16 will now be considered and the different steps in obtaining the synthesis equations for it will be explained.

By following the method given in the chapter for coupler link selection, it will be found that link number eight will be the coupler link. For group-1 mechanisms, the procedures Nos. 11, 12, 13, and 14 are called in sequence. Procedure No. 11 when called does the following:

1) It expresses:

$$D_N = f_1 (P_1, P_N, T8N, D_1)$$

$$K_N = f_2 (P_1, P_N, T8N, K_1)$$

2)  $A_N = f_3 (D_1, D_N, T1N, A_1)$

$$B_N = f_4 (D_1, D_N, T1N, B_1)$$

$$C_N = f_5 (D_1, D_N, T1N, C_1)$$

3)  $J_N = f_6 (A_1, A_N, T3N, J_1)$

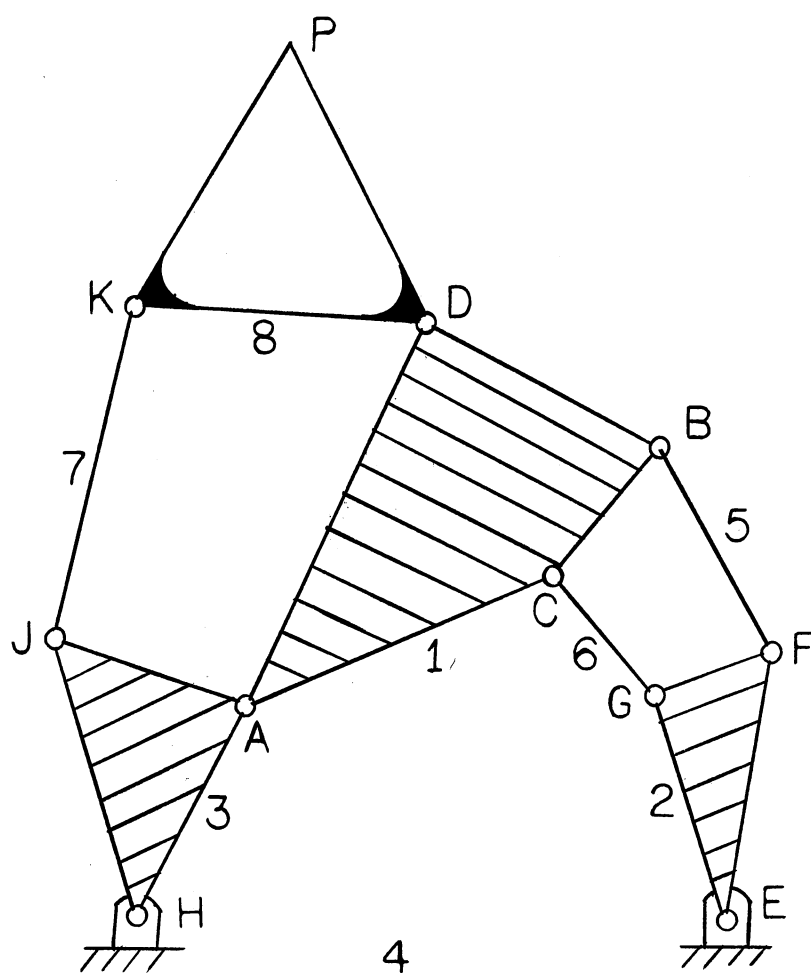


Figure 16. Group-1 Mechanism

Procedure No. 12 when called expresses the coordinates of points G and F in terms of the coordinates of E and the angle T2N using the rotation matrix, i.e.,

$$F_N = f_7 (E_1, E_1, T2N, F_1)$$

$$G_N = f_8 (E_1, E_1, T2N, G_1)$$

Procedure No. 12 when called prints out the constraining equations for all the binary links in the mechanism. This will not consider the coupler link if it happens to be a binary link. Thus, constraining equations for links JK, BF, and CG will be printed.

Lastly, Procedure No. 14 will be called and this procedure will print out the constraining equations for the ground-adjacent link. Thus, sides JH and AH will become the constraining equations for the ground-adjacent links HJA. There are five constraining equations and four unknown angles which show the equations to be consistent.

The computer output for mechanism #2 of Figure 7 is given on the following pages. It indicates the link number which has been selected as the coupler link and also gives the synthesis equations on a separate page. A brief explanation on how to interpret the synthesis equations is also given.

COMPUTER OUTPUT FOR THE EIGHT LINK  
MECHANISM OF FIGURE 7

\*\*\*\*\*

THE COUPLER LINK OF THE MECHANISM IS LINK# 4

\*\*\*\*\*

\*\*\*\*\*

THE FOLLOWING ARE THE SYNTHESIS EQUATIONS.

---

$$XGN = XPN + \cos(T4N) XG1 - \sin(T4N) YG1 - \cos(T4N) XP1 + \sin(T4N) YP1$$

$$YGN = YPN + \sin(T4N) XG1 + \cos(T4N) YG1 - \sin(T4N) XP1 - \cos(T4N) YP1$$

$$XHN = XPN + \cos(T4N) XH1 - \sin(T4N) YH1 - \cos(T4N) XP1 + \sin(T4N) YP1$$

$$YHN = YPN + \sin(T4N) XH1 + \cos(T4N) YH1 - \sin(T4N) XP1 - \cos(T4N) YP1$$

$$XAN = XHN - \cos(T6N) XH1 + \sin(T6N) YH1 + \cos(T6N) XA1 - \sin(T6N) YA1$$

$$YAN = YHN - \sin(T6N) XH1 - \cos(T6N) YH1 + \sin(T6N) XA1 + \cos(T6N) YA1$$

$$XKN = XHN - \sin(T6N) YK1 - \cos(T6N) XH1 + \sin(T6N) YH1 + \cos(T6N) XK1$$

$$YKN = YHN + \cos(T6N) YK1 - \sin(T6N) XH1 - \cos(T6N) YH1 + \sin(T6N) XK1$$

$$XCN = XKN + \sin(T7N) YK1 + \cos(T7N) XC1 - \sin(T7N) YC1 - \cos(T7N) XK1$$

$$YCN = YKN - \cos(T7N) YK1 + \sin(T7N) XC1 + \cos(T7N) YC1 - \sin(T7N) XK1$$

$$XEN = XKN + \sin(T7N) YK1 + \cos(T7N) XE1 - \sin(T7N) YE1 - \cos(T7N) XK1$$

$$YEN = YKN - \cos(T7N) YK1 + \sin(T7N) XE1 + \cos(T7N) YE1 - \sin(T7N) XK1$$

$$XJA = XKN + \cos(T7N) XJ1 - \sin(T7N) YJ1 + \sin(T7N) YK1 - \cos(T7N) XK1$$

-----  
 T7N ) XK1  
 -----

YJN = YKN + SIN ( T7N ) XJ1 + COS ( T7N ) YJ1 - COS ( T7N ) YK1 - SIN ( T7N ) XK1  
 -----

C = ( - XB1 + XAN )<sup>2</sup> - ( - XB1 + XA1 )<sup>2</sup> - ( - YB1 + YA1 )<sup>2</sup> + ( - YB1 + YAN )<sup>2</sup>  
 -----

C = ( - XD1 + XCN )<sup>2</sup> - ( - XD1 + XC1 )<sup>2</sup> - ( - YD1 + YC1 )<sup>2</sup> + ( - YD1 + YCN )<sup>2</sup>  
 -----

C = ( - XF1 + XEN )<sup>2</sup> - ( - XF1 + XE1 )<sup>2</sup> + ( - YF1 + YEN )<sup>2</sup> - ( - YF1 + YE1 )<sup>2</sup>  
 -----

O = ( XJN - XGN )<sup>2</sup> - ( XJ1 - XG1 )<sup>2</sup> - ( YJ1 - YG1 )<sup>2</sup> + ( YJN - YGN )<sup>2</sup>  
 -----

\*\*\*\*\*

THE MAXIMUM NUMBER OF PRECISION POSITIONS THAT CAN BE OBTAINED FOR  
 PCINT PATH GENERATION = 21 AND FOR RIGID BODY GUIDANCE= 11



\*\*\*\*\*

-----  
 HERE IS HOW TO INTERPRET THE DIFFERENT TERMS APPEARING IN  
 -----  
 THE SYNTHESIS EQUATIONS WHICH ARE PRINTED ON THE PREVIOUS PAGE.  
 -----

FOR EXAMPLE THE SYNTHESIS EQUATIONS MAY CONTAIN ALGEBRAIC TERMS SUCH AS  
 $XA1$ ,  $YBN$ ,  $T5N$  ETC, THEN THEY ARE TO BE INTERPRETED AS FOLLOWS

$XA1$  - X CO-OD OF THE PIVOT POINT A WHEN THE MECHANISM IS IN  
 ITS FIRST POSITION.

$YBN$  - Y CO-OD OF THE PIVOT POINT B WHEN THE MECHANISM IS IN  
 ITS N TH POSITION.

$T5N$  - ANGLE THETA WHICH THE LINK# 5 HAS UNDERGONE WHEN THE  
 MECHANISM HAS MOVED FROM POSITION 1 TO POSITION N.

P IS OFCOURSE NOT A PIVOT POINT OF THE MECHANISM BUT IS A POINT ON THE  
 PLANE OF THE COUPLER LINK.

\*\*\*\*\*

## CHAPTER VII

### SUMMARY AND CONCLUSIONS

This work eliminates the trouble of the kinematician in trying to figure out a way of obtaining the synthesis equations. A computer program is written which will take care of any single degree planar mechanism having a maximum of up to eight links. The program, when given the proper information about the mechanism, prints out the synthesis equations. These equations can be used for either point path generation or for rigid body guidance type of problems. When the equations are to be used for the point path generation problem, the angle associated with the coupler link will not be known for all the different positions of the mechanism. The requirement for this type of problem is to synthesize a mechanism so that a point on the coupler plane can trace a curve passing through the given precision points. This angle associated with the coupler link will be known in all the different positions when the problem being solved is a rigid body guidance type.

A method is also developed by which the optimum link is selected as the coupler link. If a mechanism has more than one dyad present in it, then a message is printed out regarding the same. After having obtained the equations solving them is purely a mathematical problem. There still does not exist a very sophisticated method for solving these highly non-linear synthesis equations. In the future, if such a method is developed, then it is expected that the work reported in this thesis will be of great value to the practicing designers.

## BIBLIOGRAPHY

- (1) Soni, A. H., and Lee Harrisberger. "Wanted: A Mechanism Information Research Centre." Mechanical Engineering, Vol. 91, No. 7 (July, 1969), 30-34.
- (2) Suh, C. H., and C. W. Radcliffe. "Synthesis of Plane Linkages With Use of Displacement Matrix." Trans. ASME, Journal of Engineering for Industry, Vol. 89 (1967), 206-214.
- (3) Kohli, D., and Soni, A. H., "Synthesis of Seven-link Mechanisms." Trans. ASME, Journal of Engineering for Industry, Vol. 95 (May, 1973), 533-540.
- (4) Kim, H. S., S. Hamid, and A. H. Soni. "Synthesis of Sin-Link Mechanisms for Point Path Generation." Journal of Mechanisms, Vol. 6, No. 4 (1971), 447-461.
- (5) Coutant, R. W., and A. H. Soni. "Synthesis of Stephenson Type II Sin Link Function Generator for Finitely and Infinitesimally Separated Positions." Presented at the thirteenth ASME Mechanisms Conference, New York, October 6-10, 1974.
- (6) Siddhanty, M. N., and A. H. Soni. "Simultaneous Coordination of Two Rigid Body Motions Through Infinitesimally Separated Positions." Paper No. 74-DET-90, presented at the Mechanisms Conference, New York, October, 6-10, 1974.
- (7) Hamid, S., and A. H. Soni. "Synthesis of an Eight Link Mechanism for Varieties of Motion Programs." Journal of Engineering for Industry, Trans. ASME, Vol. 95 (August, 1973), 744-750.
- (8) Quist, F. F. Jr. "Structural Synthesis and Analysis of Kinematic Chains Using Path Matrices." (M. S. Thesis, Oklahoma State University, 1971.)
- (9) Hamid, S. "Synthesis of Eight Link Mechanisms for a Variety of Motion Programs." (M. S. Thesis, Oklahoma State University, 1972.)
- (10) Huang, M. "Application of Linear and Non-linear Graphs in Structural Synthesis of Kinematic Chains." (Ph.D. Thesis, Oklahoma State University, 1972.)

- (11) Hain, K. Applied Kinematics. New York: McGraw-Hill, 1967.
- (12) Soni, A. H. "Synthesis of Multiloop Eight Link Mechanisms." Research Proposal. Stillwater, Oklahoma: Oklahoma State University. (Submitted to N. S. F., February, 1972.)
- (13) Bates, F., and M. L. Douglas. Programming Language / One, 2nd ed. New Jersey: Prentice-Hall, Inc., 1970.
- (14) PL/I - Formac Symbolic Mathematics Interpreter. Program Order Number, 360D - 03.3.004, IBM Corporation, New York, 1971.

## APPENDIX

### LISTING OF THE COMPUTER PROGRAM

```

INPUT TC FORMAC PREPROCESSOR
FRCG:PROCEDURE OPTIONS(MAIN);
FORMAC_OPTIONS;
OPTSET (LINELENGTH=72);
/*****
*
* THIS PROGRAM IS USED FOR OBTAINING THE SYNTHESIS
* EQUATIONS FOR FOUR OR ANY SIX OR EIGHT LINK PLANAR
* MECHANISM FOR THE FOLLOWING TYPES OF MCTICN PROGRAMS.
* (1) POINT PATH GENERATION AND
* (2) RIGID BODY GUIDANCE.
*
* FOR INPUT TO THIS PROGRAM REFER TO
*
* MASTERS THESIS OF
*
* SYED A. AZEEZ
*
* OKLAHOMA STATE UNIVERSITY
*
* 1976
*
*****/
DCL (I,J,II,NL,GL,CL,NLP,LUP,KEY,DYAD,ST21,ST22,KOUNT,SPCL,
LUP1,SAVE2(2) ) BIN FIXED ;
DCL (MECH,LOOP,DIAG,GIVE,LUPCOM ) (*,*) CTL BIN FIXED;
DCL (LINK,NELEM,NELCM) (*) CTL BIN FIXED;
DCL (XC1,YC1,XCN,YCN ) (8,8) CHAR(3);
DCL XCOD1(10) CHAR(3) INIT ('XA1','XB1','XC1','XD1','XE1',
'XF1','XG1','XH1','XJ1','XK1' );
DCL YCOD1(10) CHAR(3) INIT ('YA1','YB1','YC1','YD1','YE1',
'YF1','YG1','YH1','YJ1','YK1' );
DCL XCODN(10) CHAR(3) INIT ('XAN','XBN','XCN','XDN','XEN',
'XFN','XGN','XHN','XJN','XKN');
DCL YCODN(10) CHAR(3) INIT ('YAN','YBN','YCN','YDN','YEN',
'YFN','YGN','YHN','YJN','YKN');
DCL ANGLE(8) CHAR(3) INIT('T1N','T2N','T3N','T4N','T5N','T6N',
'T7N','T8N');
GET LIST (NL,GL,NLP);
ALLOCATE MECH(NL,NL),DIAG(NL,NL),GIVE(NL,NL),LCOF(NLP,NL),
LUPCOM(NLP,NL),LINK(NL),NELEM(NLP),NELCM(NLP) ;
MECH=0;
LOOP=0;
GET LIST(NELEM);
L1:DO I= 1 TO NLP;
DO J=1 TO NELEM(I);
GET LIST(LOOP(I,J));
END L1;
GET LIST (LINK);
GET DATA (MECH);
DIAG=MECH;
II=0;
L2:DO I=1 TO NL-1;
DO J=I+1 TO NL;
IF MECH(I,J)=1 THEN
CO;
II=II+1;
MECH(J,I)=1;
XC1(I,J) = XCOD1(II);
XC1(J,I) = XCOD1(II);

```

```

YCI(I,J) = YCOD1(II);
YCI(J,I) = YCOD1(II);
XCN(I,J) = XCODN(II);
XCN(J,I) = XCODN(II);
YCN(I,J) = YCCDN(II);
YCN(J,I) = YCCDN(II);
END L2;
GIVE=MECH;
IF NL=4 THEN
DO;
CL=LOOP(1,3);
CALL PROC24;
CALL PROC11;
MECH=GIVE;
CALL PROC13;
GO TO L9;
END;
CALL PROC1(L10,L11);
CALL PROC24;
IF NL < 8 THEN
L3:DO;
CALL PROC11;
MECH=GIVE;
CALL PROC12;
MECH=GIVE;
CALL PROC13;
MECH=GIVE;
CALL PROC14;
GO TO L9;
END L3;
L4:DO I=1 TO NL;
IF LINK(I)=4 THEN GO TO L3;
END L4;
IF LINK(CL)=3 THEN GO TO L3;
IF LINK(GL)=2 THEN
L5:DO;
CALL PROC15;
MECH=GIVE;
IF SPCL=0 THEN CALL PROC16;
ELSE
L6:DO;
LINK(SPCL)=2;
CALL PROC11;
MECH=GIVE;
CALL PROC12;
MECH=GIVE;
LINK(SPCL)=3;
CALL PROC13;
MECH=GIVE;
CALL PROC17;
GO TO L9;
END L6;
IF SPCL = 0 THEN GO TO L6;
MECH=GIVE;
L7:DO I=1 TO NL;
IF MECH(GL,I)=1 THEN
IF LINK(I)=2 THEN
DO;
CALL PROC18;
GO TO L9;

```

```

END L7;
GO TO L3;
END L5;
KOUNT=0;
L8:DC I=1 TO NL;
  IF MECH(GL,I)=1 THEN
    IF LINK(I)=2 THEN KOUNT=KOUNT+1;
  END L8;
  IF KOUNT = 3 THEN GO TO L3;
  CALL PROC19;
L9:FP=3*(NL-1);
  RB=(3*NL-2)/2;
  PUT EDIT
  ('*****')
  (SKIP(3),X(1),A)
  ('*****')(A)
  ('THE MAXIMUM NUMBER OF PRECISION POSITIONS THAT CAN BE OBTAIN')
  (SKIP(2),COL(5),A)
  ('NED FOR')(A)
  ('POINT PATH GENERATION =',PP)(SKIP(2),COL(5),A,F(3))
  ('AND FOR RIGID BODY GUIDANCE=',RB)(A,F(3));
  PUT EDIT
  ('*****')
  (PAGE,SKIP(12),COL(30),A)
  ('*****')(A)
  ('HERE IS HOW TO INTERPRET THE DIFFERENT TERMS APPEAR')
  (SKIP(5),COL(33),A)('ING IN')(A)
  ('-----')
  (COL(33),A)('-----')(A)
  ('THE SYNTHESIS EQUATIONS WHICH ARE PRINTED ON THE PREVIOUS')
  (SKIP(2),COL(30),A)('EVIOUS PAGE.')(A)
  ('-----')
  (COL(30),A)('-----')(A)
  ('FOR EXAMPLE THE SYNTHESIS EQUATIONS MAY CONTAIN ALGEBRAIC TERMS')
  (SKIP(5),COL(30),A)('TERMS SUCH AS')(A)
  ('XA1,YBN,T5N ETC, THEN THEY ARE TO BE INTERPRETED AS FOLLOWS')
  (SKIP(2),COL(30),A)
  ('XA1 - X CO-ORD OF THE PIVOT POINT A WHEN THE MECHANISM IS IN ITS')
  (SKIP(2),COL(33),A)('ITS FIRST POSITION.')(SKIP(2),COL(33),A)
  ('YBN - Y CO-ORD OF THE PIVOT POINT B WHEN THE MECHANISM IS IN ITS')
  (SKIP(2),COL(33),A)('ITS N TH POSITION.')(SKIP(2),COL(33),A)
  ('T5N - ANGLE THETA WHICH THE LINK# 5 HAS UNDERGONE WHEN THE')
  (SKIP(2),COL(33),A)
  ('MECHANISM HAS MOVED FROM POSITION 1 TO POSITION N.')(A)
  (SKIP(2),COL(33),A)
  ('P IS OFCOURSE NOT A PIVOT POINT OF THE MECHANISM BUT IS A POINT')
  (SKIP(2),COL(30),A)('POINT ON THE')(A)
  ('PLANE OF THE COUPLER LINK.')(SKIP(2),COL(30),A)
  ('*****')
  (SKIP(5),COL(30),A)
  ('*****')(A);
F1:FORMAT(SKIP(2),COL(40),A);
GO TO ENDPGM;
L10:PUT EDIT
  ('*****')
  (PAGE,SKIP(19),R(F1))
  ('THE MECHANISM SELECTED FOR THE JOB IS IN TRUTH A')
  (SKIP(4),R(F1))
  ('PSEUDO',NL,' LINK MECHANISM.SELECT A MECHANISM WHICH')
  (SKIP,R(F1),F(2),A)

```



```

('WILL NOT HAVE MORE THAN ONE DYAD PRESENT IN IT.')(R(F1))
('*****')
(SKIP(3),R(F1),PAGE);
GO TO ENDPGM;
L11:PUT EDIT
('*****')
(PAGE,SKIP(13),COL(40),A)
('THE MECHANISM HAS AN INCIPENDENT LOOP FORMING A')
(SKIP(3),R(F1))
('FOUR BAR AND REMOVAL OF THE DYAD FORMING THIS FOUR')(R(F1))
('BAR LOOP WILL STILL LEAVE THE MECHANISM WITH A')(R(F1))
('SINGLE DEGREE OF FREEDOM. IT IS THEREFCRE A PSEUDG')(R(F1))
('EIGHT LINK MECHANISM.ANY OTHER TRUE EIGHT LINK')(R(F1))
('MECHANISM IS TO BE SELECTED BEFCRE SYNTHESIS')(R(F1))
('EQUATIONS CAN BE OBTAINED.')(R(F1))
('*****')
(SKIP(3),R(F1),PAGE);
PROC1:PROCEDURE(LAB1,LAB2);
/*****
*
*          PROCEDURE# 1
*
*  THIS PROCEDURE SELECTS THE COUPLER LINK.
*
*****/
DCL (LAB1,LAB2) LABEL;
CALL PROC2;
IF DYAD > 1 THEN GO TO LAB1;
IF DYAD = 1 THEN CALL PROC3;
IF CL > 0 THEN
  DO;
L1:DO I=1 TO NLP;
  DO J=1 TO NELEM(I);
  IF LCCP(I,J)=CL THEN
    IF NELEM(I)=4 THEN GO TO LAB2;
  END L1;
GO TO END_P1;
END;
LUF=NLP;
LUF1=NLP;
NELCM(1)=NELEM(1);
LUPCOM(1,*)=LOOP(1,*);
KOUNT=1;
L2:CALL PROC4;
ST21=SAVE2(1);
ST22=SAVE2(2);
IF ST21 > ST22 THEN CALL PROC5;
CALL PROC6;
IF KEY < LUP1 THEN CALL PRCC7;
IF LUP >= 2 THEN GO TO L2;
CALL PROC8;
CALL PROC9;
IF DYAD=1 THEN CALL PROC10;
END_P1:END PROC1;
PROC2:PROCEDURE;
/*****
*
*          PROCEDURE# 2
*
*  DETERMINES THE TCTAL NUMBER OF DYADS PRESENT IN THE

```

```

* MECHANISM.
*
*****/
DCL (I,J,II,STORE(2) EXTERNAL ) BIN FIXED;
CL=0;
DYAD=0;
L1: DO I=1 TO NL-1;
  IF GL=I | LINK(I) /= 2 THEN GO TO END_L1;
  II=I+1;
L2: DO J = II TO NL;
  IF DIAG(I,J) /= 1 | LINK(J) /= 2 THEN GO TO END_L2;
  DYAD=DYAD+1;
  STORE(1)=I;
  STORE(2)=J;
END_L2: END L2;
END_L1: END L1;
END PROC2;
PROC3: PROCEDURE;
/*****
*
* PROCEDURE# 3
*
* CHECKS IF A BINARY LINK OF THE DYAD IS CONNECTED TO THE
* GROUND LINK AND IF SO THE OTHER BINARY LINK IS SELECTED AS
* THE COUPLER LINK.
*
*****/
DCL STORE(2) EXTERNAL BIN FIXED;
IF MECH(GL,STORE(1))=1 THEN CL = STORE(2);
ELSE
IF MECH(GL,STORE(2)) = 1 THEN CL = STORE(1);
ELSE CL = 0;
END PROC3;
PROC4: PROCEDURE;
/*****
*
* PROCEDURE# 4
*
* DETERMINES THE LOOP NUMBER WHICH WILL HAVE AT LEAST TWO
* LINKS IN COMMON WITH THE FIRST LOOP. THE GROUND LINK IS
* NOT CONSIDERED.
*
*****/
DCL (I,J,K,COM,SAVE1(2) EXTERNAL ) BIN FIXED;
L1: DO K = 2 TO LUP;
  COM=0;
  DO I = 2 TO NELCM(KOUNT);
L2: DO J = 1 TO NELEM(K);
  IF LUPCCM(KCUNT,I) /= LOOP(K,J) THEN GO TO L3;
  COM = COM+1;
  SAVE1(COM) = I;
  SAVE2(COM) = J;
  IF COM /= 2 THEN GO TO L3;
  KEY=K;
  RETURN;
L3: END L2;
END L1;
END PROC4;
PROC5: PROCEDURE;
/*****

```

```

*
*                                     PROCEDURE# 5
*
* REARRANGES THE LOOP DETERMINED IN PROCEDURE FOUR IF NEEDED
*
*****/
DCL (I,COUNT,DUM ) BIN FIXED;
COUNT = NELEM(KEY);
L1:DUM = LOOP(KEY,1);
IF ST22 = 1 THEN
DO;
DO I=1 TO COUNT-1;
LOOP(KEY,I) = LOOP(KEY,I+1);
END;
LOCP(KEY,COUNT) = DUM;
ST21 = ST21-1;
ST22 = COUNT;
END;
ELSE DO;
DO I = 1 TO COUNT-1;
LOOP(KEY,I) = LOCP(KEY,I+1);
END;
LOCP(KEY,COUNT) = DUM;
ST21 = ST21-1;
ST22 = ST22-1;
GO TO L1;
END;
END PROC5;
PROC6:PROCEDURE;
/*****
*
*                                     PROCEDURE# 6
*
* COMBINES THE MOST RECENTLY COMBINED LOOP WITH THE LOOP
* DETERMINED IN PROCEDURE FOUR.
*
*****/
DCL (COUNT,COUNT1,ST11,DUM,KP1,NEK,POST,SAVE1(2) EXTERNAL )
BIN FIXED;
COUNT=0;
KP1 = KOUNT+1;
ST11 = SAVE1(1);
NEK = NELCM(KOUNT);
DO I = 1 TO ST11;
COUNT = COUNT+1;
LUPCOM(KP1,COUNT) = LUPCOM(KOUNT,I);
END;
L1:DO I = ST21+1 TO NELEM(KEY);
COUNT = COUNT+1;
DUM = LOOP(KEY,I);
LUPCOM(KP1,COUNT) = DUM;
IF MECH(GL,DUM) = 1 THEN GO TO L3;
COUNT1=0;
DO J = ST11+1 TO NEK;
COUNT1 = COUNT1+1;
IF LUPCOM(KOUNT,J) = DUM THEN GO TO L2;
END L1;
L2:POST = ST11+1+COUNT1;
DO I = POST TO NEK;
COUNT = COUNT+1;

```

```

      LUPCOM(KP1,COUNT) = LUPCOM(KOUNT,I);
      END;
L3:NELCM(KP1) = COUNT;
      LUP = LUP-1;
      KOUNT=KOUNT+1;
      END PROC6;
PROC7:PROCEDURE;
  /*****
  *
  *                               PROCEDURE# 7
  *
  *   WILL UPDATE THE LOOPS WHEN INVOKED BY A CALL STATEMENT.
  *
  *****/
  DCL (I,J,N) BIN FIXED;
  LUP1=LUP-1;
L1:DO I = KEY TO LUP1;
      N = NELEM(I+1);
      NELEM(I)=N;
      DO J = 1 TO N;
        LOCP(I,J) = LOOP(I+1,J);
      END L1;
      END PROC7;
PRCC8:PRCCEDURE;
  /*****
  *
  *                               PROCEDURE# 8
  *
  *   DETERMINES WHICH OF THE COMBINED LOOP HAS MAXIMUM NUMBER
  *   OF LINKS IN IT.
  *
  *****/
  DCL (I,TEMP1,TEMP2,BIGLUP EXTERNAL ) BIN FIXED;
  TEMP1 = NELCM(1);
  BIGLUP = 1;
L1:DO I = 2 TO NLP;
      TEMP2 = NELCM(I);
      IF TEMP1 > TEMP2 THEN GO TO L2;
      BIGLUP = I;
      TEMP1 = TEMP2;
L2:END L1;
      END PROC8;
PROC9:PROCEDURE;
  /*****
  *
  *                               PROCEDURE# 9
  *
  *   SELECTS THE COUPLER LINK FROM THE LCCP DETERMINED IN
  *   PROCEDURE EIGHT.
  *
  *****/
  DCL (A,C,D,TEMP1,TEMP2,BIGLUP EXTERNAL ) BIN FIXED,B DEC FLOAT;
  A = NELCM(BIGLUP);
  B = FLOAT(A)/2.0;
  A = A/2;
  C = A+1;
  D=C+1;
  IF B = FLOAT(A) THEN CL = LUPCOM(BIGLUP,C);
  ELSE DO;
    TEMP1 = LUPCOM(BIGLUP,C);
  
```

```

TEMP2 = LUPCOM(BIGLUP,0);
A=LINK(TEMP1);
C=LINK(TEMP2);
IF A >= C THEN CL=TEMP2;
ELSE CL=TEMP1;
END;
END PROC9;
PRCC10:PROCEDURE;
/*****
*
*                               PROCEDURE# 10
*
* WILL MAKE A FINAL CHECK ONLY WHEN A DYAD IS PRESENT IN
* THE MECHANISM TO SEE THAT THE COUPLER LINK IS PROPERLY
* SELECTED.
*
*****/
DCL (I,J,COUNT,( STORE(2) EXTERNAL)) BIN FIXED,
(ONE,TWO,MID) DEC FLOAT;
L1:DO I = 1 TO NLP;
COUNT = NELCM(I);
DO J = 1 TO COUNT;
IF LUPCOM(I,J) = STORE(1) THEN GO TO L2;
END L1;
L2:MID = FLOAT(COUNT)/2.0;
ONE = ABS(MID-FLOAT(J));
TWO = ABS(MID-FLOAT(J+1));
IF ONE >= TWO THEN CL = LUPCOM(I,J+1);
ELSE CL = LUPCOM(I,J);
END PROC10;
PRCC11:PROCEDURE;
/*****
*
*                               PROCEDURE# 11
*
* WILL EXPRESS THE COORDINATES OF THE REQUIRED PIVOT POINTS
* IN TERMS OF PROPER QUANTITIES USING THE DISPLACEMENT
* MATRIX.
*
*****/
DCL (I,J,K,L,II,JJ,KK,COUNT1,COUNT2,COUNT3,(NOTE1,NOTE2,
NOTE3)(5),(CHECK1,CHECK2(2),CHECK3(2)) EXTERNAL ) BIN FIXED;
COUNT1 = 0;
CHECK1 = 0;
L1:DO I = 1 TO NL;
IF MECH(CL,I) = 0 THEN GO TO END_L1;
IF LINK(I) = 2 THEN
DC;
COUNT1 = COUNT1+1;
NOTE1(COUNT1) = I;
END;
MECH(CL,I) = 0;
MECH(I,CL) = 0;
CALL PROC21(XCN(CL,I),YCN(CL,I),XC1(CL,I),YC1(CL,I),'XPN','YPN',
'XP1','YP1',ANGLE(CL));
END_L1:END L1;
IF COUNT1 = 0 THEN GO TO END_P11;
L2:DO I = 1 TO COUNT1;
II = NOTE1(I);
COUNT2 = 0;

```

```

L3: DO J = 1 TO NL;
    IF MECH(II,J) = 0 THEN GO TO END_L3;
    IF J = GL THEN
        DO;
            CHECK1 = CHECK1+1;
            CHECK2(CHECK1) = II;
            CHECK3(CHECK1) = CL;
            GO TO END_L3;
        END;
    IF LINK(J) = 2 THEN
        DO;
            COUNT2 = COUNT2+1;
            NOTE2(COUNT2) = J;
        END;
    MECH(II,J) = 0;
    MECH(J,II) = 0;
    CALL PROC21(XCN(II,J),YCN(II,J),XC1(II,J),YC1(II,J),XCN(II,CL),
        YCN(II,CL),XC1(II,CL),YC1(II,CL),ANGLE(II));
END_L3: END L3;
    IF COUNT2 = 0 THEN GO TO END_L2;
L4: DO J = 1 TO COUNT2;
    JJ = NOTE2(J);
    COUNT3 = 0;
L5: DO K = 1 TO NL;
    IF MECH(JJ,K) = 0 THEN GO TO END_L5;
    IF K = GL THEN
        DO;
            CHECK1 = CHECK1+1;
            CHECK2(CHECK1) = JJ;
            CHECK3(CHECK1) = II;
            GO TO END_L5;
        END;
    IF LINK(K) = 2 THEN
        DO;
            COUNT3 = COUNT3+1;
            NOTE3(COUNT3) = K;
        END;
    MECH(JJ,K) = 0;
    MECH(K,JJ) = 0;
    CALL PROC21(XCN(JJ,K),YCN(JJ,K),XC1(JJ,K),YC1(JJ,K),XCN(II,JJ),
        YCN(II,JJ),XC1(II,JJ),YC1(II,JJ),ANGLE(JJ));
END_L5: END L5;
    IF COUNT3 = 0 THEN GO TO END_L4;
L6: DO K = 1 TO COUNT3;
    KK = NOTE3(K);
L7: DO L = 1 TO NL;
    IF MECH(KK,L) = 0 THEN GO TO END_L7;
    IF L = GL THEN
        DO;
            CHECK1 = CHECK1+1;
            CHECK2(CHECK1) = KK;
            CHECK3(CHECK1) = JJ;
            GO TO END_L7;
        END;
    CALL PROC21(XCN(KK,L),YCN(KK,L),XC1(KK,L),YC1(KK,L),XCN(KK,JJ),
        YCN(KK,JJ),XC1(KK,JJ),YC1(KK,JJ),ANGLE(KK));
END_L7: END L7;
    END L6;
END_L4: END L4;
END_L2: END L2;

```

```

END_P11:ENC PROC11;
PRCC12:PRCCEDURE;
/*****
*
*                               PROCEDURE# 12
*
*   WILL EXPRESS THE COORDINATES OF THE REQUIRED PIVOT POINTS
*   IN TERMS OF PROPER QUANTITIES USING THE ROTATION MATRIX.
*
*****/
DCL (I,J,K,L,M,KK,LL,MM,NN,COUNT1,COUNT2,NOTE1(5),NOTE2(5))
BIA FIXED;
L1:DC I = 1 TO NL;
COUNT1=0;
COUNT2=0;
IF MECH(GL,I) = 1 THEN
IF LINK(I) = 2 THEN GO TO END_L1;
ELSE DO;
MECH(GL,I)=0;
MECH(I,GL)=0;
L2:DO J = 1 TO NL;
IF MECH(I,J) = 1 THEN
IF J = CL THEN GO TO END_L1;
ELSE DO;
COUNT1 = COUNT1+1;
NOTE1(COUNT1) = J;
MECH(I,J) = 0;
MECH(J,I) = 0;
IF LINK(J) = 2 THEN GO TO END_L2;
L3:DC K = 1 TO NL;
IF MECH(J,K) = 1 THEN
IF LINK(K) = 2 | K = CL THEN GO TO END_L1;
ELSE DO;
COUNT2 = COUNT2+1;
NOTE2(COUNT2) = K;
END L3;
END_L2:END L2;
L4:DO L = 1 TO COUNT1;
JJ = NOTE1(L);
CALL PROC20(XCN(I,JJ),YCN(I,JJ),XC1(I,JJ),YC1(I,JJ),XC1(I,GL),
YC1(I,GL),ANGLE(I));
END L4;
KK = 0;
L5:DO M = 1 TO COUNT1;
LL = NOTE1(M);
MM = LINK(LL);
IF MM = 2 THEN GO TO END_L5;
L6:DO N = 1 TO MM-1;
KK = KK+1;
NN = NOTE2(KK);
CALL PROC21(XCN(LL,NN),YCN(LL,NN),XC1(LL,NN),YC1(LL,NN),
XCN(LL,I),YCN(LL,I),XC1(LL,I),YC1(LL,I),ANGLE(LL));
END L6;
END_L5:END L5;
END_L1:END L1;
END PROC12;
PRCC13:PRCCEDURE;
/*****
*
*                               PROCEDURE# 13
*

```

```

*
* PRINTS OUT CONSTRAINING EQUATIONS FOR ALL THE BINARY
* LINKS PRESENT IN THE MECHANISM EXCEPT FOR THE COUPLER
* LINK IF IT HAPPENS TO BE A BINARY LINK.
*
*****/
DCL (I,J,K,L,M,COUNT,NOTE(2)) BIN FIXED;
L1:DC I = 1 TO NL;
  IF LINK(I) = 2 THEN
    IF I = CL | I = GL THEN GO TO END_L1;
  ELSE DO;
    COUNT = 0;
  L2:DO J = 1 TO NL;
    IF MECH(I,J) = 1 THEN
      DO;
        COUNT = COUNT+1;
        NOTE(COUNT) = J;
        IF COUNT = 2 THEN GO TO L3;
      END L2;
  L3:K = NOTE(1);
    L = NOTE(2);
    IF K = GL THEN GO TO L4;
    IF L = GL THEN GO TO L5;
    M = K; K = L; L = M;
  L4:CALL PROC23(XCN(I,L),YCN(I,L),XC1(I,L),YC1(I,L),XC1(I,K),
    YC1(I,K),'0');
    GO TO END_L1;
  L5:CALL PROC22(XCN(I,L),YCN(I,L),XC1(I,L),YC1(I,L),XCN(I,K),
    YCN(I,K),XC1(I,K),YC1(I,K),'0');
END_L1:END L1;
END PROC13;
PROC14:PROCEDURE;
  /*****
  *
  * PROCEDURE# 14
  *
  * PRINTS OUT CONSTRAINING EQUATIONS FOR GROUND-ADJACENT
  * LINKS.
  *
  *****/
  DCL (I,J,K,L,(CHECK1,CHECK2(2),CHECK3(2)) EXTERNAL) BIN
  FIXED;
  IF CHECK1 = 0 THEN GO TO END_P14;
  L1:DO I = 1 TO CHECK1;
    J = CHECK2(I);
    IF LINK(J) > 3 THEN
      DO;
        K = CHECK3(I);
        MECH(J,K) = 0;
        MECH(K,J) = 0;
      END;
    MECH(J,GL) = 0;
    MECH(GL,J) = 0;
    DO L = 1 TO NL;
      IF MECH(J,L) = 1 THEN
        CALL PROC23(XCN(J,L),YCN(J,L),XC1(J,L),YC1(J,L),XC1(J,GL),
          YC1(J,GL),'0');
    END L;
  END L1;
END_P14:END PROC14;
PROC15:PROCEDURE;

```



```

/*****
*
*                               PROCEDURE# 15
*
*   CHECKS IF THE MECHANISM GIVEN IS A GROUP-2 MECHANISM.
*
*****/
DCL (I,J,K,COUNT1,CCUNT2) BIN FIXED;
SPCL = 0;
L1:DO I = 1 TO NL;
  COUNT2 = 0;
  IF MECH(GL,I) = 1 THEN
    IF LINK(I) = 2 THEN GO TO END_L1;
  ELSE DO;
    MECH(GL,I) = 0;
    MECH(I,GL) = 0;
  L2:DO J=1 TO NL;
    IF MECH(I,J) = 1 THEN
      IF J = CL | LINK(J) = 2 THEN GO TO L3;
    ELSE DO;
      MECH(I,J) = 0;
      MECH(J,I) = 0;
      COUNT1 = 0;
      DO K = 1 TO NL;
        IF MECH(J,K) = 1 THEN
          DO;
            IF LINK(K)=2 THEN
              IF K = CL THEN COUNT1=COUNT1+1;
            IF COUNT1 = 2 THEN GO TO L3;
            COUNT2 = COUNT2+1;
            IF COUNT2 = 4 THEN GO TO L4;
          END L2;
        L3:MECH=GIVE;
      END_L1:END L1;
      GO TO END_P15;
    L4:SPCL=I;
  END_P15:END PROC15;
PRCC16:PROCEDURE;
/*****
*
*                               PROCEDURE# 16
*
*   CHECKS IF THE MECHANISM GIVEN IS A GROUP-3 MECHANISM.
*
*****/
DCL (I,J,COUNT) BIN FIXED;
L1:DO I = 1 TO NL;
  COUNT = 0;
  IF MECH(CL,I) = 1 THEN
    IF LINK(I) = 2 THEN
      DO;
        MECH(CL,I) = 0;
        MECH(I,CL) = 0;
        DO J = 1 TO NL;
          IF MECH(I,J) = 1 THEN
            IF LINK(J) = 2 THEN
              DO;
                COUNT = COUNT+1;
                IF CCUNT = 2 THEN GO TO L2;
              END L1;
            END L1;
          END L1;
        END L1;
      END L1;
    END L1;
  END L1;

```

```

      GO TO END_P16;
L2: MECH=GIVE;
      SPCL=1;
END_P16: END PROC16;
PRCC17: PROCEDURE;
      /*****
      *
      *                               PROCEDURE# 17
      *
      * PRINTS OUT CONSTRAINING EQUATIONS FOR THE NECESSARY
      * TERNARY LINK.
      *
      *****/
      DCL (I,J,K,KK,CCUNT,NCTE(2)) BIN FIXED;
      COUNT=0;
      I = SPCL;
      IF MECH(GL,I) = 1 THEN
      DO;
      MECH(I,GL) = 0;
      MECH(GL,I) = 0;
L1: DO J = 1 TO NL;
      IF MECH(I,J) = 1 THEN
      DO;
      COUNT = COUNT+1;
      NOTE(COUNT) = J;
      IF CCUNT = 2 THEN GO TO L2;
      GO TO END_L1;
L2: DO K = 1 TO COUNT;
      KK = NOTE(K);
      CALL PROC23(XCN(I,KK),YCN(I,KK),XC1(I,KK),YC1(I,KK),XC1(I,GL),
      YC1(I,GL),'0');
      END L2;
      J = NOTE(1);
      K = NOTE(2);
      CALL PROC22(XCN(I,J),YCN(I,J),XC1(I,J),YC1(I,J),XCN(I,K),
      YCN(I,K),XC1(I,K),YC1(I,K),'0');
      GO TO END_P17;
END_L1: END L1;
      END;
      ELSE
L3: DO;
      MECH(CL,I) = 0;
      MECH(I,CL) = 0;
      DO J = 1 TO NL;
      IF MECH(I,J) = 1 THEN
      DO;
      COUNT = COUNT+1;
      NOTE(COUNT) = J;
      IF CCUNT = 2 THEN GO TO L4;
      END L3;
L4: DO K = 1 TO COUNT;
      KK = NOTE(K);
      CALL PROC22(XCN(CL,I),YCN(CL,I),XC1(CL,I),YC1(CL,I),XCN(I,KK),
      YCN(I,KK),XC1(I,KK),YC1(I,KK),'0');
      END L4;
      J = NOTE(1);
      K = NOTE(2);
      CALL PROC22(XCN(I,J),YCN(I,J),XC1(I,J),YC1(I,J),XCN(I,K),
      YCN(I,K),XC1(I,K),YC1(I,K),'0');
END_P17: END PROC17;

```

PRCC18:PROCEDURE;

```

/*****
*
*                               PROCEDURE# 18
*
*   WILL BE CALLED IF MECHANISM# 1 OF GROUP-4 IS GIVEN.
*
*****/
DCL (I,J,K,II,JJ,COUNT,NOTE(5)) BIN FIXED;
L1:DO I = 1 TO NL;
  IF MECH(GL,I) = 1 THEN
    IF LINK(I) = 2 THEN
      DO;
        MECH(GL,I) = 0;
        MECH(I,GL) = 0;
        DO J = 1 TO NL;
          IF MECH(I,J) = 1 THEN
            IF LINK(J) = 2 THEN GO TO L2;
        END L1;
      L2:LINK(J) = 2;
        MECH(I,GL) = 1;
        MECH(GL,I) = 1;
        CALL PROC11;
        MECH=GIVE;
        CALL PROC12;
        MECH=GIVE;
        COUNT = 0;
      L3:DO I = 1 TO NL;
        IF MECH(J,I) = 1 THEN
          DO;
            COUNT = COUNT+1;
            NOTE(COUNT) = I;
            IF COUNT = 3 THEN GO TO L4;
          END L3;
        L4:DO I = 1 TO 2;
          II = NOTE(I);
          DO K = I+1 TO COUNT;
            JJ = NOTE(K);
            CALL PROC22(XCN(J,II),YCN(J,II),XC1(J,II),YC1(J,II),XCN(J,JJ),
              YCN(J,JJ),XC1(J,JJ),YC1(J,JJ),'0');
          END L4;
          LINK(J) = 3;
          CALL PROC13;
        END PROC18;

```

PROC19:PROCEDURE;

```

/*****
*
*                               PROCEDURE# 19
*
*   WILL BE CALLED IF MECHANISM# 2 OF GROUP-4 IS GIVEN.
*
*****/
DCL (I,J,K,L,M) BIN FIXED;
L1:DO I=1 TO NL;
  IF MECH(CL,I)=1 THEN
    DO;
      MECH(CL,I)=0;
      MECH(I,CL)=0;
      DO J=1 TO NL;
        IF MECH(I,J)=1 THEN

```

```

      IF LINK(J) = 2 THEN GO TO L2;
    END L1;
  L2: MECH=GIVE;
      LINK(J)=2;
      CALL PROC11;
      MECH=GIVE;
      MECH(I,J)=0;
      MECH(J,I)=0;
  L3: DO K=1 TO NL;
      IF MECH(J,K)=1 THEN
        IF LINK(K)=2 THEN
          DO;
            CALL PROC21(XCN(K,J),YCN(K,J),XC1(K,J),YC1(K,J),XCN(I,J),
              YCN(I,J),XC1(I,J),YC1(I,J),ANGLE(J));
            L=K;
          END;
        ELSE M=K;
      END L3;
      DO K=1 TO 2;
        CALL PROC22(XCN(L,J),YCN(L,J),XC1(L,J),YC1(L,J),XCN(J,M),
          YCN(J,M),XC1(J,M),YC1(J,M),'0');
        L=I;
      END;
      LINK(J)=3;
      CALL PROC13;
    END PROC19;
  PRCC20: PROCEDURE(XBN,YBN,XB1,YB1,XQ1,YQ1,ALPHA);
    /*****
    *
    *          PROCEDURE# 20
    *
    *  WILL BE CALLED WHEN THE ROTATION MATRIX IS TO BE USED.
    *
    *****/
    DCL (XBN,YBN) CHAR(100) VARYING,(XB1,YB1,XQ1,YQ1,ALPHA)
      CHAR(3);
    LET ("XBN"=XB1*CCS("ALPHA")-YB1*SIN("ALPHA")+XQ1-XQ1*
      COS("ALPHA")+YQ1*SIN("ALPHA");
      "YBN"=XB1*SIN("ALPHA")+YB1*COS("ALPHA")+YQ1-YQ1*COS
      ("ALPHA")-XQ1*SIN("ALPHA"));
    PRINT_OUT("XBN";"YBN");
    ATCMIZE("XBN";"YBN");
  END PROC20;
  PROC21: PROCEDURE(XAN,YAN,XA1,YA1,XPN,YPN,XP1,YP1,THETA);
    /*****
    *
    *          PROCEDURE# 21
    *
    *  WILL BE CALLED WHEN THE DISPLACEMENT MATRIX IS TO BE USED.
    *
    *****/
    DCL (XAN,YAN) CHAR(100) VARYING,(XA1,YA1,XPN,YPN,XP1,YP1,THETA)
      CHAR(3);
    LET ("XAN"=XA1*CCS("THETA")-YA1*SIN("THETA")+XPN-XP1*
      COS("THETA")+YP1*SIN("THETA");
      "YAN"=XA1*SIN("THETA")+YA1*COS("THETA")+YPN-XP1*SIN
      ("THETA")-YP1*COS("THETA"));
    PRINT_OUT("XAN";"YAN");
    ATCMIZE("XAN";"YAN");
  END PROC21;

```

```

PROC22: PROCEDURE(XAN,YAN,XA1,YA1,XBN,YBN,XB1,YB1,C);
/*****
*
*                               PROCEDURE# 22
*
*   IS CALLED WHENEVER A CONSTRAINING EQUATION IS TO BE
*   WRITTEN BETWEEN TWO POINTS BOTH OF WHICH CAN MOVE ALONG
*   WITH THE MECHANISM.
*
*****/
DCL (XAN,YAN,XA1,YA1,XBN,YBN,XB1,YB1) CHAR(3),
C CHAR(100) VARYING;
LET("O"=("XAN"- "XBN")**2+ ("YAN"- "YBN")**2- ("XA1"- "XB1")**2
- ("YA1"- "YB1")**2);
PRINT_OUT("O");
ATOMIZE("O");
END PROC22;

PROC23: PROCEDURE(XAN,YAN,XA1,YA1,XB1,YB1,C);
/*****
*
*                               PROCEDURE# 23
*
*   IS CALLED WHENEVER A CONSTRAINING EQUATION IS TO BE
*   WRITTEN BETWEEN TWO POINTS WHEN ONE OF THEM LIES ON THE
*   GROUND LINK.
*
*****/
DCL (XAN,YAN,XA1,YA1,XB1,YB1) CHAR(3),C CHAR(100) VARYING;
LET("O"=("XAN"- "XB1")**2+ ("YAN"- "YB1")**2- ("XA1"- "XB1")**2
- ("YA1"- "YB1")**2);
PRINT_OUT("O");
ATOMIZE("O");
END PROC23;

PROC24: PROCEDURE;
/*****
*
*                               PROCEDURE# 24
*
*   PRINTS OUT THE NUMBER OF THE COUPLER LINK.
*
*****/
PUT EDIT ('*****')
(PAGE,SKIP(25),COL(43),A)
('THE COUPLER LINK OF THE MECHANISM IS LINK#',CL)
(SKIP(3),COL(43),A,F(2))
('*****')
(SKIP(3),COL(43),A)
('*****')
(PAGE,SKIP(3),X(1),A)
('*****')(A)
('THE FOLLOWING ARE THE SYNTHESIS EQUATIONS.')
(SKIP(3),COL(15),A)
('-----')
(COL(16),A)('')(SKIP(3),A);
END PROC24;
ENDPGM:END PROG;

```

VITA

Syed Abdul Azeez

Candidate for the Degree of  
Master of Science

**Thesis:** AN ALGORITHM FOR OBTAINING SYNTHESIS EQUATIONS FOR ANY (MAXIMUM EIGHT LINKS) SINGLE DEGREE PLANAR MECHANISM FOR PATH GENERATION AND RIGID BODY GUIDANCE TYPES OF MOTION PROGRAMS

**Major Field:** Mechanical Engineering

**Biographical:**

**Personal Data:** Born in Nizamabad, India, March 3, 1950, the son of Ayesha Effendi Begum and Dr. Syed Abdul Lateef.

**Education:** Studied up to tenth class in St. George's Grammar School, Hyderabad, India, and graduated from People's Tutorial College, Hyderabad, India; received Bachelor of Engineering degree in Mechanical Engineering from Osmania University, Hyderabad, India, in 1973; completed the requirements for the Degree of Master of Science at Oklahoma State University in May, 1976.

**Professional Experience:** Working part time as a teaching and research assistant at Oklahoma State University from Spring, 1974, to Fall, 1975.